

01170

5  
2º

**Universidad Nacional Autónoma de México  
Facultad de Ingeniería  
División de Estudios de Posgrado**

**Estimación de movimiento y segmentación basada en  
movimiento en secuencias de imágenes**

Tesis presentada para obtener el grado de  
Maestro en Ingeniería Eléctrica  
(Comunicaciones)

**Daniel Gatica Pérez**

Ciudad Universitaria, junio de 1996

**TESIS CON  
FALLA DE ORIGEN**

**TESIS CON  
FALLA DE ORIGEN**



Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

## Resumen

Dos de los objetivos del análisis de movimiento en secuencias de imágenes son la *estimación* y la *segmentación de movimiento aparente*, procesos íntimamente relacionados, y valiosos tanto en visión computacional como en procesamiento de secuencias de imágenes. Este trabajo se dedicó al análisis, estudio e implantación de ambas tareas, utilizando para la evaluación secuencias sintéticas y reales. La estrategia general desarrollada fue la siguiente: (1) estimar un *campo denso de vectores de desplazamiento* a partir de dos tramas de una secuencia, y (2) emplear el campo generado como información principal para efectuar la *partición* de la imagen en *regiones con movimiento similar*. La estimación del desplazamiento se realizó con *algoritmos pel-recursivos*. Se estudiaron e implantaron algunos de los más representativos, y se realizó un análisis de su desempeño, considerando como criterios principales de evaluación la calidad de reconstrucción de las secuencias, y la congruencia del campo estimado con la realidad física. Se encontró que uno de los métodos, basado en teoría de estimación de Wiener, produce campos útiles para una segmentación subsecuente. Para la implantación de esta tarea, se propuso un algoritmo basado en *campos aleatorios de Markov* y *estimación bayesiana* (criterio *máximo a posteriori*), que consiste en la estimación simultánea de un campo de *etiquetas de segmentación* caracterizado por un modelo de movimiento lineal simplificado, y de un campo de *líneas de discontinuidad de movimiento*; el algoritmo integra dentro de un mismo modelo información de diferente naturaleza (campo de movimiento, intensidad de la secuencia, medidas de confianza acerca de la estimación del desplazamiento, bordes espaciales) y conocimiento a priori acerca del problema. Los resultados obtenidos con el método propuesto son satisfactorios para las diferentes secuencias empleadas.

*A mis padres*

# Gracias

- A mi familia, por haber estado siempre aquí.
- A mi asesor, Francisco García Ugalde, por las enseñanzas, la oportunidad de trabajar con él, y el estímulo constante.
- A Víctor García Garduño, por las horas de discusión y la paciencia.
- A los profesores que participaron en mi formación, en especial a Rogelio Alcántara Silva, Abel Herrera Camacho, Fernando Lepe Casillas y Jorge Lira Chávez.
- A Boris Escalante Ramírez, Miguel Moctezuma Flores y Jesús Savage Carmona, por haber aceptado formar parte de mi jurado y por su apoyo.
- A todos mis amigos de la sección de Eléctrica, por estos años. En particular, a Oscar Viveros Cancino y a Rubisel Tovilla Heredia por las conversaciones nocturnas y por los programas proporcionados.

# Indice

<b>1</b>	<b>Introducción.</b>	<b>4</b>
1.1	¿Para qué analizar el movimiento en una secuencia de imágenes? . . .	5
1.2	Objetivos del trabajo y estrategia utilizada. . . . .	6
1.3	Estructura de la tesis. . . . .	8
<b>I</b>	<b>Estimación de movimiento aparente en secuencias de imágenes utilizando algoritmos pel-recursivos</b>	<b>10</b>
<b>2</b>	<b>Métodos de estimación de movimiento en secuencias de imágenes: estado del arte.</b>	<b>11</b>
2.1	Movimiento 2D y movimiento aparente. . . . .	11
2.2	¿Por qué es difícil medir el movimiento en una secuencia de imágenes? . . . . .	13
2.3	Clasificación de los métodos de estimación de movimiento aparente. . . . .	15
2.3.1	Clasificación según el modelo de movimiento. . . . .	15
2.3.2	Clasificación según la estrategia de medición del movimiento. . . . .	18
2.4	Conclusiones. . . . .	22
<b>3</b>	<b>Los algoritmos pel-recursivos de estimación de movimiento.</b>	<b>23</b>
3.1	Introducción. . . . .	23
3.1.1	La diferencia de trama desplazada (DFD). . . . .	23
3.2	El algoritmo pel-recursivo básico. . . . .	24
3.3	El algoritmo de Walker y Rao. . . . .	27
3.4	El algoritmo basado en estimación de Wiener. . . . .	28
3.5	Lógica de los algoritmos pel-recursivos de estimación de movimiento. . . . .	31
3.6	Compensación de movimiento. . . . .	32
3.7	Variaciones de los algoritmos pel-recursivos. . . . .	33
3.7.1	Algoritmos combinados WR-BD. . . . .	33
3.7.2	Métodos a múltiples resoluciones (MR). . . . .	34
3.8	Conclusiones. . . . .	36
<b>4</b>	<b>Implantación y evaluación de los algoritmos pel-recursivos.</b>	<b>37</b>
4.1	Criterios de evaluación de los algoritmos. . . . .	37
4.2	Secuencias de prueba. . . . .	38

4.3	Resultados. . . . .	43
4.3.1	Movimiento translacional puro. . . . .	43
4.3.2	Movimiento sintético variado. . . . .	46
4.3.3	Secuencia real Miss América. . . . .	57
4.3.4	Secuencia real Interview. . . . .	63
4.3.5	Recapitulación . . . . .	70
4.3.6	Consideraciones para la implantación de los algoritmos. . . . .	72
4.3.7	Limitaciones de los métodos. . . . .	78
4.3.8	Aplicaciones. . . . .	79
4.3.9	Conclusiones. . . . .	82

## **II Segmentación basada en movimiento utilizando campos aleatorios de Markov 83**

<b>5</b>	<b>Segmentación basada en movimiento aparente en secuencias de imágenes. <span style="float: right;">84</span></b>
5.1	Introducción. . . . . 84
5.1.1	¿Para qué segmentar una secuencia de imágenes en base al movimiento? . . . . . 85
5.1.2	Aspectos particulares de una SBM. . . . . 86
5.2	Métodos para SBM en una secuencia de imágenes. . . . . 87
5.2.1	Métodos directos de SBM. . . . . 88
5.2.2	Métodos indirectos de SBM. . . . . 89
5.3	Discusión. . . . . 93
<b>6</b>	<b>Método propuesto de segmentación basada en movimiento. <span style="float: right;">95</span></b>
6.1	Introducción. . . . . 95
6.2	Planteamiento general del problema. . . . . 95
6.2.1	Campos de Markov y distribución de Gibbs. . . . . 95
6.2.2	Estimación bayesiana global usando MRF. . . . . 97
6.2.3	Aplicación de la teoría a una SBM. . . . . 98
6.3	Modelado de las funciones de energía. . . . . 101
6.3.1	Modelo de las observaciones. . . . . 102
6.3.2	Modelo a priori de la segmentación. . . . . 103
6.4	Esquema de optimización. . . . . 106
6.4.1	Inicialización. . . . . 107
6.4.2	Etiquetado y numerado de regiones. . . . . 108
6.4.3	Estimación de parámetros. . . . . 108
6.4.4	Esquema de relajación ICM modificado. . . . . 108
6.4.5	Criterio de finalización del algoritmo. . . . . 108
6.5	Diferencias entre el algoritmo de Murray y Buxton y el algoritmo propuesto. . . . . 109
6.6	Conclusiones. . . . . 109

<b>7</b>	<b>Resultados y evaluación del algoritmo de segmentación basada en movimiento.</b>	<b>111</b>
7.1	Selección de los parámetros del modelo. . . . .	111
7.2	Resultados y discusión. . . . .	112
7.2.1	Secuencia Seqmilt. . . . .	113
7.2.2	Secuencia Miss América. . . . .	120
7.2.3	Secuencia Interview. . . . .	123
7.2.4	Recapitulación. . . . .	126
7.3	Perspectivas del trabajo. . . . .	126
7.3.1	Mejoras al algoritmo. . . . .	126
7.3.2	Criterios de fusión de regiones. . . . .	128
7.3.3	Posibles aplicaciones. . . . .	129
7.4	Conclusiones. . . . .	132
<b>8</b>	<b>Conclusiones generales.</b>	<b>133</b>
<b>A</b>	<b>Configuraciones de los cliques del campo de líneas.</b>	<b>135</b>
<b>B</b>	<b>Referencias Bibliográficas.</b>	<b>137</b>

# Capítulo 1

## Introducción.

Una *secuencia de imágenes digitales* es una serie ordenada de imágenes representada por una función de intensidad o luminancia  $I(x, y, t)$ , definida sobre una rejilla discreta tridimensional  $L_3$ . Las secuencias de imágenes pueden adquirirse, por ejemplo, por medio de una cámara de televisión, por rayos X, por ultrasonido, o sintetizarse como en el caso de animación. El creciente uso de los servicios de video y el desarrollo de la ciencia y la tecnología en las comunicaciones y la computación han llevado a la aparición de sistemas de procesamiento y análisis de secuencias de imágenes en áreas tan diversas como la industria del entretenimiento, los servicios de comunicaciones visuales, los servicios de presentación múltiple (multimedios), la educación, la medicina, el control a distancia, y la investigación científica.

El interés en este campo condujo al surgimiento de varias áreas especializadas, en particular, del *procesamiento* y del *análisis* de secuencias de imágenes. El primero se refiere al conjunto de operaciones de filtrado, interpolación y/o submuestreo espaciotemporal, y compresión, con los objetivos de mejorar la calidad visual de las imágenes, convertir las secuencias a distintos formatos de video, y reducir la cantidad de datos para su transmisión efectiva, respectivamente. El segundo trata sobre las operaciones realizadas sobre las secuencias con el propósito de obtener datos que permitan su representación o interpretación.

El movimiento relativo entre los objetos de una escena y el sistema de adquisición de imágenes da lugar al *movimiento aparente* de aquéllos en la secuencia de imágenes. Durante los últimos quince años, las áreas del procesamiento de secuencias de imágenes y de visión computacional (interactuando dentro de la rama del análisis de secuencias de imágenes conocida como *análisis de movimiento*) se han interesado en el problema del movimiento de los objetos en una secuencia. ¿Por qué? Porque el movimiento proporciona una valiosa fuente de información acerca del ambiente de la escena y de la estructura tridimensional de los objetos que la conforman, útil tanto para los sistemas biológicos como para los computacionales. Algunas razones más específicas para cada caso se detallan enseguida.

## 1.1 ¿Para qué analizar el movimiento en una secuencia de imágenes?

El interés de las ciencias biológicas en la relación entre la visión y el movimiento no es nuevo, puesto que la visión ha sido considerada siempre como el sentido más poderoso [HILD87]. Se ha descubierto que la detección, la medición y la interpretación del movimiento juegan un papel crucial en los sistemas de visión de gran cantidad de animales, incluido el hombre, en operaciones básicas para la supervivencia, como la detección y persecución de una presa, la detección y reconocimiento de un depredador, y la circulación veloz en un ambiente tridimensional complejo. También se descubrió que muchos animales dotados de sistemas de visión cuentan con receptores de movimiento específicos en su corteza visual [FRAN91]. En el hombre, algunas de las tareas primordiales son la detección de objetos en movimiento (base de toda la percepción del movimiento), el reconocimiento y la segmentación de objetos basándose en su movimiento, y la obtención de la estructura tridimensional de la escena (información de profundidad) a partir de la proyección bidimensional que produce una superficie móvil sobre el ojo [MARR82].

El uso extensivo del movimiento en los sistemas biológicos ha influenciado muchas ramas de la tecnología. Actualmente, para la ingeniería, el interés por el movimiento en secuencias de imágenes es diverso. Para el desarrollo de sistemas de visión computacional, el estudio del movimiento es básico para realizar algunas de las operaciones fundamentales: reconocer y seguir objetos, desplazarse entre ellos, y estimar cuantitativamente sus propiedades dinámicas. Estos sistemas encuentran aplicaciones variadas [HUAN81], [HUAN83], [HORN86], [AGGA88], [FRAN91]:

- **Robótica y automatización.** En un ambiente visual controlado, un sistema de visión hace más versátil a un robot manipulador, permitiéndole que tolere variaciones en la posición y la orientación de las partes a ensamblar o tomar, por ejemplo. Mucho más elaborado debe ser un sistema que permita a un robot navegar de forma autónoma o seguir objetos.

- **Control de tráfico.** Un sistema de visión fijo puede ser de gran utilidad en carreteras para control en casetas de cobro y verificación de límites de velocidad.

- **Navegación autónoma.** Un sistema de visión puede auxiliar al conductor como copiloto, o en un futuro llegar a sustituirlo.

- **Aplicaciones militares.** El reconocimiento de blancos, y el seguimiento automático de proyectiles son dos tristes ejemplos del uso de esta información.

- **Aplicaciones biomédicas.** El análisis de secuencias de imágenes del corazón se usa como apoyo en el diagnóstico y tratamiento de enfermedades cardiovasculares. Otros usos se dan en radiología y en medicina nuclear, o en el seguimiento de células sanguíneas.

- **Percepción remota.** Existen fenómenos atmosféricos (como el movimiento de nubes) y oceanográficos que pueden modelarse e interpretarse mediante análisis de movimiento.

El análisis de movimiento encuentra muchas aplicaciones también en procesamiento de secuencias de imágenes, específicamente en tareas de:

- **Compresión.** La idea de explotar la redundancia temporal existente en una secuencia de imágenes a través de la información de movimiento, permite reducir el volumen de la información a transmitir. Existe una amplia variedad de algoritmos de compresión, conocidos como técnicas de compresión por *estimación y compensación de movimiento* [MUSM85], [LI94], [AIZA95], [CHIA95]. Los servicios involucrados con estas técnicas son numerosos: videotelefonía, videoconferencia, televisión de alta definición, correo electrónico multimedia, percepción remota, periódicos electrónicos, bases de datos multimedia interactivas, juegos de video, telemedicina, y generación interactiva de imágenes por computadora, entre otros.

- **Reducción de ruido.** La presencia de ruido en una secuencia de imágenes degrada tanto su calidad visual como la efectividad de los algoritmos de procesamiento subsecuentes. Por tanto, los métodos de reducción de ruido no sólo aumentan la calidad visual de una secuencia, sino que incrementan potencialmente el desempeño de funciones posteriores, como codificación, análisis o interpretación [BRAI95]. Las técnicas de filtrado temporales y espacio-temporales con compensación de movimiento se utilizan con frecuencia con este objetivo.

- **Restauración.** La restauración se refiere a la perfilación de una secuencia de imágenes degradada por difuminaciones ópticas, que ocurren cuando un solo punto de un objeto se esparce sobre varios pixels de la imagen, lo cual puede presentarse debido al movimiento relativo entre la cámara y la escena, o por una adquisición de la secuencia con la cámara desenfocada. Existen métodos multitrama para restauración, que hacen uso de la correlación temporal y de la información de movimiento existente entre tramas [TEKA95].

- **Conversión de estándares de video digital.** Esta función permite el intercambio de información entre sistemas de video con formatos de almacenamiento, transmisión y despliegue distintos; puede incluir operaciones de interpolación o submuestreo de la señal de video [SEZA93], [TEKA95]. Los esquemas de conversión de formatos con compensación de movimiento son óptimos para ciertos casos.

## 1.2 Objetivos del trabajo y estrategia utilizada.

El análisis de movimiento incluye en su dominio, funciones como la *detección* y la *estimación* del movimiento aparente en el plano imagen, la *segmentación* de la

escena basada en movimiento, el *reconocimiento* de patrones y su *seguimiento* a lo largo del tiempo, la *extracción* de la estructura y del movimiento tridimensional de los objetos presentes en la escena, y la *interpretación* de la misma.

En particular, la *estimación de movimiento (EM)* y la *segmentación basada en movimiento (SBM)* son dos problemas apasionantes, fuertemente relacionados, pero que pueden encontrar aplicaciones independientes. El **objetivo general** de este trabajo fue estudiar estas dos tareas básicas del análisis de movimiento, y útiles tanto en visión computacional como en procesamiento de secuencias de imágenes.

Existe en la literatura una abundante variedad de familias de métodos para estimar el movimiento aparente en una secuencia de imágenes, que se diferencian por varios aspectos, mencionados en la primera parte de este trabajo. De ellas, se estudió en detalle e implantó una, conocida genéricamente como *algoritmos pel-recursivos*; dichos algoritmos calculan un *campo denso de vectores de desplazamiento aparente* a partir de un par de imágenes (*tramas*) consecutivas de la secuencia. Fueron desarrollados para compresión de secuencias de imágenes, y han evolucionado durante los últimos años. La evaluación de los algoritmos, con respecto a la calidad del campo estimado y a la calidad de reconstrucción de las secuencias, se realizó utilizando secuencias sintéticas y reales.

En el documento se muestra que el movimiento estimado utilizando un algoritmo pel-recursivo específico (basado en estimación de Wiener) presenta buenas características cualitativas y cuantitativas, que lo hacen apto para extraer de él información de nivel superior. Una idea que surge entonces de manera natural es utilizar dicho campo para una aplicación distinta a la que ha recibido tradicionalmente (compresión de secuencias por codificación predictiva y compensación de movimiento). En este trabajo se emplea como base para implantar la segunda función deseada, una segmentación basada en movimiento.

Para ello, se revisaron distintos métodos que efectúan tal tarea; este estudio llevó a la elección de un enfoque probabilístico, basado en *campos aleatorios de Markov (MRF)*, para obtener la solución. La formalización markoviana ha demostrado su eficacia en otras aplicaciones dentro del procesamiento de imágenes, y resulta apropiada por las ventajas que ofrece para las características particulares de una segmentación de campo de desplazamiento.

El planteamiento de un modelo general de una SBM utilizando *estimación bayesiana*, de acuerdo al criterio *MAP (máximo a posteriori)*, y MRF condujo a la propuesta de un algoritmo, que puede verse como una extensión de un método conocido, pero cuyo modelo es más rico en información, a fin de explotar de mejor manera el conocimiento a priori que se tiene del problema, y manteniendo al método suficientemente flexible para su aplicación a distintos tipos de secuencias. Al final, el algoritmo propuesto se implantó y evaluó con imágenes artificiales y reales, con la finalidad de analizar sus ventajas y limitaciones.

Resumiendo, los **objetivos particulares** planteados para este trabajo fueron tres:

a) Analizar una familia de métodos diferenciales de estimación de movimiento aparente, conocidos como *algoritmos pel-recursivos*, evaluar su desempeño y estudiar sus posibles aplicaciones.

b) Mostrar que es factible realizar una *segmentación basada en el movimiento* estimado por un método pel-recursivo.

c) Proponer un método que realice dicha segmentación.

La Figura 1.1. muestra el esquema de la estrategia global empleada.

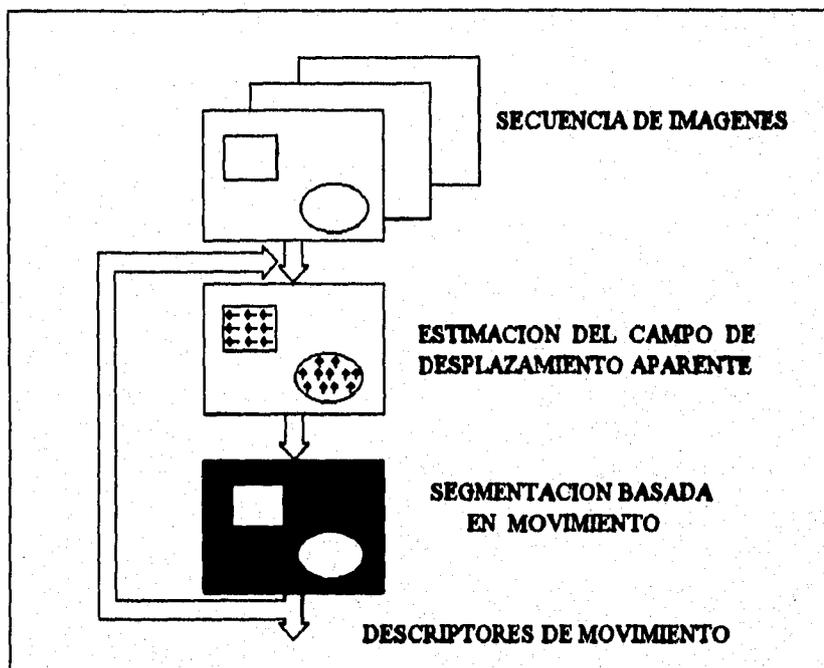


Figura 1.1: Estrategia general de trabajo utilizada.

### 1.3 Estructura de la tesis.

Este reporte se divide en dos partes principales, una para cada tarea estudiada. Los propósitos de los primeros tres capítulos son: definir claramente el problema de la estimación del movimiento en una secuencia de imágenes, y clasificar los diferentes enfoques para resolverlo (capítulo 2); describir con precisión a la familia específica de métodos pel-recursivos (capítulo 3), y mostrar los resultados obtenidos con su

implantación en computadora para imágenes sintéticas y reales, junto con su análisis y evaluación (capítulo 4).

Los siguientes tres capítulos se ocupan de la segmentación basada en movimiento. En orden se presentan: la definición del problema, la comparación entre las múltiples maneras de abordarlo, y la discusión que condujo a la elección del enfoque markoviano (capítulo 5); la descripción detallada del algoritmo de SBM propuesto (capítulo 6), y su evaluación con distintos tipos de secuencias de imágenes (capítulo 7).

Las conclusiones generales y las perspectivas del trabajo aparecen al final del documento.

Más allá de pretensiones y lugares comunes, mi intención final es transmitir el entusiasmo que esta área de estudio despertó en mí desde el inicio de este proyecto - y que se ha ido acrecentando con el tiempo - no sólo por la naturaleza misma de los problemas a resolver, sino por el gran número de aplicaciones actuales y futuras que este campo de la ciencia encuentra.

## **Parte I**

# **Estimación de movimiento aparente en secuencias de imágenes utilizando algoritmos pel-recursivos**

## Capítulo 2

# Métodos de estimación de movimiento en secuencias de imágenes: estado del arte.

### 2.1 Movimiento 2D y movimiento aparente.

En general, el análisis del movimiento de una secuencia de imágenes se puede dividir en dos partes: la primera es la medición o estimación del movimiento, esto es, la asignación de magnitud y dirección al desplazamiento o a la velocidad de los elementos de la imagen, basándose en los cambios en los patrones de intensidad de la secuencia, con lo que se genera un *campo de vectores de movimiento*. La segunda etapa consiste en el empleo de la información obtenida en el primer paso para realizar funciones más elaboradas, como por ejemplo, separar la imagen en distintas regiones (segmentación). Finalmente, la utilización de esta nueva información puede tener aplicaciones en procesamiento de secuencias (codificación, restauración) o en visión computacional (reconstrucción o interpretación de la escena).

Una secuencia de imágenes real se forma mediante la proyección de escenas tridimensionales (3D) en un plano bidimensional (2D), conocido como *plano imagen*. Mientras que en la escena real, el movimiento 3D puede caracterizarse en términos de la *velocidad* o el *desplazamiento* 3D instantáneos de cada uno de los puntos de la escena, en la secuencia todo el movimiento ocurre en el plano imagen. Antes de hablar de los métodos de estimación, conviene definir adecuadamente algunos términos básicos.

**Movimiento 2D y movimiento aparente.** El primero, también conocido como *movimiento proyectado*, se refiere a la proyección (perspectiva u ortográfica) del movimiento 3D en el plano imagen [HORN86]. En cambio, el segundo es el movimiento *observado* en el plano imagen a partir de la variación en los patrones de intensidad de la secuencia. Este es el único movimiento al cual se tiene acceso en la secuencia a procesar.

Dependiendo de la cantidad cinemática a medir (desplazamiento o velocidad) se manejan de acuerdo a lo anterior, los conceptos de *campo de desplazamiento 2D* y *campo de correspondencia*, y de *velocidad 2D* y *flujo óptico*, respectivamente [TEKA95]. Esto es, los campos de desplazamiento y de velocidad 2D son proyecciones de los campos 3D respectivos en el plano imagen, mientras que el campo de correspondencia y el flujo óptico son las funciones de desplazamiento y velocidad, que se pueden percibir al observar, en la secuencia de imágenes, los patrones de intensidad 2D variantes en el tiempo (sobre una rejilla espacio-temporal  $L_3$ ). Es decir, son el desplazamiento y la velocidad *aparentes*.

Así, el problema de estimación de movimiento aparente puede plantearse de dos formas: (a) estimar los vectores de correspondencia utilizando *dos tramas*

$$\mathbf{d}(x, y, t; k\Delta t) = (d_x(x, y, t; k\Delta t), d_y(x, y, t; k\Delta t)) \quad (2.1)$$

entre los tiempos  $t - k\Delta t$  y  $t$  y para toda  $(x, y, t)$  en la rejilla espaciotemporal  $L_3$ , donde  $k$  es un entero y  $\Delta t$  es el intervalo entre tramas; o (b) estimar los vectores de flujo óptico

$$\mathbf{v}(x, y, t; k\Delta t) = (v_x(x, y, t; k\Delta t), v_y(x, y, t; k\Delta t)) \quad (2.2)$$

en esta misma rejilla. El primer caso es el que interesa en este trabajo, por lo que se desea encontrar, para todos los pixels en la secuencia, la correspondencia

$$I(x, y, t) = I(x - d_x(x, y, t), y - d_y(x, y, t), t - k\Delta t) = I(\mathbf{z} - \mathbf{d}(x, y, t), t - k\Delta t) \quad (2.3)$$

donde

- $I(x, y, t)$  es la intensidad de cada pixel de la secuencia.
- $\mathbf{z} = (x, y)$  es la posición de cada pixel .
- $\mathbf{d}(x, y, t) = (d_x(x, y, t), d_y(x, y, t))$  es el vector de desplazamiento de cada pixel durante el intervalo  $(t - k\Delta t, t)$ .

En el resto de este trabajo, cuando se haga referencia al campo de desplazamiento se estará haciendo alusión al campo de vectores de correspondencia; ocurrirá algo similar para el campo de velocidad y el flujo óptico.

## 2.2 ¿Por qué es difícil medir el movimiento en una secuencia de imágenes?

Importantes problemas para el procesamiento de imágenes y la visión computacional, incluidos la estimación y segmentación del campo de movimiento aparente, y la reconstrucción de una escena 3D a partir de una secuencia de imágenes, son problemas de naturaleza inversa, es decir, intentan recuperar propiedades físicas 3D a partir de los datos 2D [BERT88], [FRAN91], [TEKA95]. En particular, la estimación de movimiento como tal (esto es, considerando sólo dos tramas de la secuencia, y sin suposiciones adicionales acerca de la naturaleza del movimiento) es un *problema mal planteado (ill-posed)* en el sentido de Hadamard [BERT88]:

- a) La solución no existe, o
- b) La solución no es única, o
- c) Las soluciones no dependen continuamente de los datos de entrada.

La estimación de movimiento viola las tres condiciones necesarias (existencia, unicidad, continuidad) para que un problema sea *bien planteado (well-posed)*:

a) En aquellas regiones de la escena que quedan cubiertas o descubiertas por los objetos en movimiento entre tramas sucesivas, no existe correspondencia (*problema de oclusión*).

b) Si se observa la ecuación (2.1), y se considera cada componente del vector de desplazamiento como una variable independiente, entonces puede verse que para cada pixel existe una sola ecuación con dos incógnitas [TEKA95]. El análisis teórico [HORN86] muestra que dada la naturaleza local de la medición del movimiento, sólo se puede calcular una componente de la velocidad: la paralela al gradiente de intensidad  $\nabla I$  (*problema de apertura*).

c) La estimación de movimiento es altamente sensible al ruido presente en la secuencia: un error pequeño en los datos puede producir un error grande en la solución [BERT88]. De ahí la necesidad de encontrar métodos de estimación que sean robustos al ruido.

Existe además un problema teórico importante: ¡los campos de movimiento aparente y proyectado en general son distintos! [VERR89]. Sólo bajo ciertas condiciones se garantiza que sean similares o que estén relacionados de alguna forma [BERT88]. Se suele hablar de este hecho como de la *ambigüedad* del campo de movimiento porque, por ejemplo, distintas condiciones de iluminación en una misma escena pueden producir movimientos aparentes diferentes, o porque la ausencia de gradientes espaciales en la imagen hace que un movimiento real no sea observable, o porque distintas superficies, experimentando movimientos reales diferentes, producen el mismo movimiento aparente. La figura 2.2 ilustra el problema general de

la estimación del movimiento aparente.

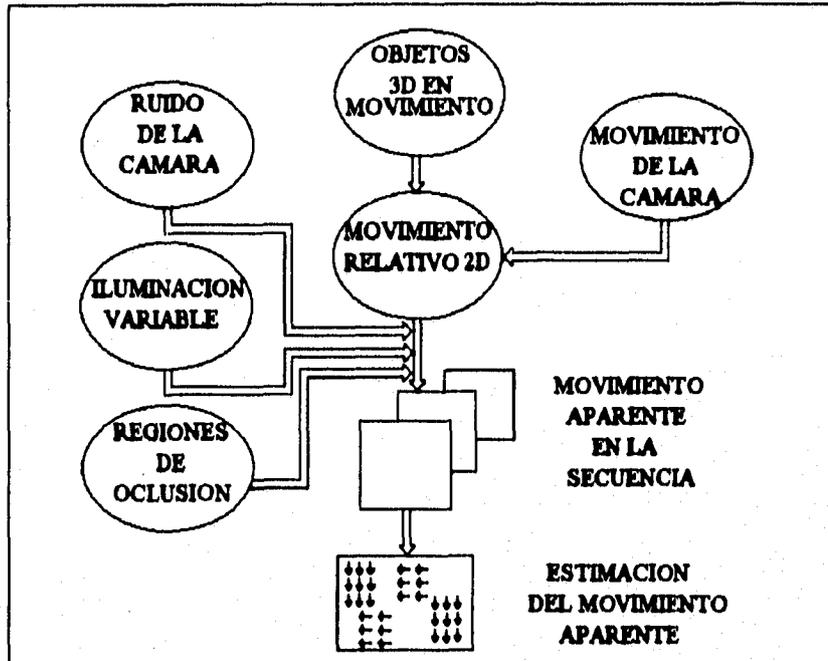


Figura 2.1: El problema de la estimación del movimiento aparente.

No obstante lo anterior, muchos de los enfoques de estimación de movimiento - clásicos y recientes - han partido del hecho de que las discrepancias entre los dos campos no son usualmente muy grandes [HORN86], en particular en regiones con gradientes de intensidad notables [HEIT93].

Por otra parte, en [VERR89] se demuestra también que los dos campos tienen las mismas propiedades cualitativas. De aquí que hayan surgido esquemas que describen el movimiento cualitativamente [FRAN91]. Puede decirse que ambos enfoques - a favor y en contra del uso cuantitativo de la información de movimiento obtenida a partir de un campo de correspondencia o de flujo óptico - gozan hoy de cabal salud, porque hay aplicaciones en donde cada uno de ellos produce buenos resultados.

Como puede verse, el problema de la estimación de movimiento no es trivial, y las formas de tratar las dificultades mencionadas son variadas. Distintas soluciones se han obtenido a través del uso de información adicional acerca de las características deseadas o supuestas para el campo de movimiento (*restricciones de suavizado*), en forma de métodos de regularización determinísticos o enfoques estocásticos de optimización global (ver secciones 2.3.2.3 y 2.3.2.4).

## 2.3 Clasificación de los métodos de estimación de movimiento aparente.

Existe un elevado número de métodos para la estimación de movimiento aparente en una secuencia de imágenes. La variedad se refleja en la diversidad de aplicaciones ya discutidas. No es de extrañar, por tanto, que en la literatura se encuentren varias clasificaciones, según distintos criterios, de forma que cada método puede pertenecer a más de una categoría. En la tabla 2.1 se muestran tres clasificaciones a menudo encontradas.

Criterio de clasificación	Clases de métodos	Métodos específicos
Complejidad de los algoritmos de visión computacional	<ul style="list-style-type: none"> <li>● bajo nivel</li> <li>● alto nivel</li> </ul>	
Modelo de movimiento	<ul style="list-style-type: none"> <li>● paramétricos</li> <li>● cuasiparamétricos</li> <li>● no paramétricos</li> </ul>	
Estrategia de medición del movimiento	<ul style="list-style-type: none"> <li>● puesta en correspondencia</li> <li>● diferenciales</li> <li>● por transformadas</li> <li>● bayesianos</li> </ul>	<ul style="list-style-type: none"> <li>● bloques de pixels</li> <li>● rasgos de imagen</li> <li>● basados en la ecuación de flujo óptico</li> <li>● pel-recursivos</li> </ul>

Tabla 2.1. Clasificación de los métodos de estimación de movimiento aparente en secuencias de imágenes.

La primera clasificación [KONR92] es bastante general, y no se comenta más en detalle. Las otras dos clasificaciones se describen a continuación.

### 2.3.1 Clasificación según el modelo de movimiento.

#### 2.3.1.1. Modelos de movimiento.

Una suposición importante para encontrar una solución al problema de la estimación del movimiento aparente es la inclusión - explícita o tácita - de algún

*modelo de movimiento*, que describa la proyección del movimiento 3D de la escena en el plano imagen [ANAN93]. El empleo de modelos de la estructura de los objetos y del campo de movimiento conduce a la obtención de mejores estimaciones.

Matemáticamente, la inclusión de un modelo de movimiento se define por una transformación  $T$  que describe el comportamiento cinemático de cada pixel individual, o de un grupo de pixels dentro de una región, entre dos tramas sucesivas de una secuencia:

$$I(\mathbf{z}, t + 1) = T(I(\mathbf{z}, t), \theta) \quad (2.4)$$

donde  $\theta$  es el *vector de parámetros* del modelo de movimiento  $\Theta_M$  que caracteriza cada pixel o región. Este conjunto de parámetros permite describir translaciones, rotaciones, divergencias, e incluso deformaciones.

Son tres los factores a considerar para la definición de un modelo: el sistema de proyección óptico de los objetos en el plano imagen (perspectivo u ortográfico), el tipo de movimiento 3D de los objetos (rígido o no rígido), y el modelo geométrico de la estructura de los objetos (conformados por superficies planas o de otro tipo). Diferentes combinaciones de estos factores dan origen a diferentes modelos, que se distinguen por dos características: *complejidad* y *compacidad*, existiendo un compromiso entre ambas.

Los modelos lineales de movimiento más utilizados son los siguientes [GARC95b]:

1. *Modelo completo de movimiento 3D rígido planar bajo proyección perspectiva* (8 parámetros).

$$\begin{bmatrix} d_x \\ d_y \end{bmatrix} = \begin{bmatrix} a_1 + a_2x + a_3y + a_7xy + a_8x^2 \\ a_4 + a_5x + a_6y + a_8xy + a_7x^2 \end{bmatrix} \quad (2.5)$$

donde  $\mathbf{d} = (d_x, d_y)$  es el vector de desplazamiento en el plano imagen,  $(x, y)$  son las coordenadas originales de cada pixel, y  $\theta = (a_1, \dots, a_8)$  son los parámetros del modelo, definidos en [ADIV85].

2. *Modelo lineal o affine* ( $\Theta_{MLIN}$ , 6 parámetros). Este modelo se obtiene al considerar solamente los términos lineales del modelo completo.

$$\begin{bmatrix} d_x \\ d_y \end{bmatrix} = \begin{bmatrix} a_1 \\ a_4 \end{bmatrix} + \begin{bmatrix} a_2 & a_3 \\ a_5 & a_6 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (2.6)$$

3. *Modelo lineal simplificado* ( $\Theta_{MLS}$ , 4 parámetros). Si se supone (a) que el movimiento de rotación sólo se presenta alrededor del eje perpendicular al plano ubicado en el centro de gravedad de cada región considerada, y (b) que todos los planos que conforman cada objeto o región de la escena son paralelos al plano imagen, el modelo completo se reduce a

$$\begin{bmatrix} d_x \\ d_y \end{bmatrix} = \begin{bmatrix} t_x \\ t_y \end{bmatrix} + \begin{bmatrix} k & -\theta \\ \theta & k \end{bmatrix} \begin{bmatrix} x - x_g \\ y - y_g \end{bmatrix} \quad (2.7)$$

donde  $(t_x, t_y)$  es el *vector de translación* del centro de gravedad de la región  $(x_g, y_g)$ ,  $k$  es el *coeficiente de divergencia* (translación a lo largo del eje óptico, y  $\theta$  es el *ángulo de rotación* alrededor del eje perpendicular al plano imagen.

4. *Modelo constante o translacional* ( $\Theta_{MC}$ ) (2 parámetros). Esta representación es la más simple posible, que en sentido estricto es un modelo no paramétrico.

$$\begin{bmatrix} d_x \\ d_y \end{bmatrix} = \begin{bmatrix} t_x \\ t_y \end{bmatrix} \quad (2.8)$$

Existen además otros modelos (cuadráticos, deformables) que no se abordan en el presente trabajo.

### 2.3.1.2. Clasificación.

Los métodos basados en *modelos paramétricos* determinan los valores de los parámetros del modelo para cada región. Pueden aplicarse de manera exacta sólo en situaciones bastante limitadas, pero representan la mejor aproximación disponible para un buen número de casos, por lo que actualmente se usan en muchas aplicaciones.

Los métodos basados en *modelos cuasiparamétricos* representan el movimiento de cada pixel mediante una combinación de una componente paramétrica que es válida para toda una región y de una componente local que varía de pixel a pixel. Han sido típicamente usados para codificación.

Los métodos basados en *modelos no paramétricos* fueron los primeros en desarrollarse, y utilizan una o varias restricciones de *suavizado* o *uniformidad* (de manera local o global), de forma tal que el campo de movimiento resultante no puede expresarse en forma cerrada por un modelo paramétrico 3-D. El campo resultante puede ser menos preciso, pero también su aplicación es más general. Han sido los más extensamente usados.

## 2.3.2 Clasificación según la estrategia de medición del movimiento.

Esta clasificación se basa en la filosofía general que guía el proceso de extracción de la información de movimiento. Existen tres grandes enfoques: (a) la *puesta en correspondencia* - entre dos o más tramas de la secuencia - de ciertos rasgos o características importantes de la escena; (b) la explotación directa de las *variaciones espacio-temporales* de la información de intensidad en la secuencia de imágenes; y (c) la utilización de *transformadas* para obtener la información de movimiento en otro espacio de representación. Existe un cuarto enfoque, basado en *métodos bayesianos*, que se ha explotado más recientemente, y que no está contenido por completo en ninguno de los tres anteriores, por lo que puede mencionarse de forma independiente. Cada uno de estos métodos se comenta con algún detalle en las subsecciones siguientes, sin intentar ser exhaustivo en la descripción; para eso pueden consultarse [FRAN91], [BARR94] o [TEKA95].

### 2.3.2.1. Métodos de estimación por puesta en correspondencia.

Estos métodos buscan alguna medida de similitud entre dos grandes clases de estructuras: *bloques de pixels* de forma y tamaño variable, y *descriptores de imagen* previamente extraídos de la secuencia (puntos, contornos, esquinas, regiones).

#### Métodos por puesta en correspondencia de bloques (*block matching*).

Se desarrollaron originalmente en los años 70 para compresión de secuencias de imágenes [MUSM85]. Actualmente, pueden considerarse como los más populares para estas tareas, ya que incluso se han adoptado para estándares internacionales de compresión de video, como H.261, MPEG1 y MPEG2 [CHIA95], debido a su baja complejidad computacional y a la simplicidad requerida para su implantación en circuitería. En ellos, se supone que cada imagen se compone de bloques que se mueven de una trama a otra de la secuencia. Por lo tanto, la estimación consiste en asociar un vector de velocidad o de desplazamiento a cada uno de dichos bloques, utilizando alguna medida de correlación. En un principio, estos métodos suponían movimiento translacional puro, pero se han desarrollado nuevas estrategias que toman en cuenta movimientos más complejos; asimismo, se trabaja para elaborar técnicas de post-procesamiento que mejoren la calidad visual de las secuencias reconstruidas, a costa de un aumento considerable en la complejidad computacional.

#### Métodos por puesta en correspondencia de rasgos de imagen (*feature-based*).

Requieren de una etapa previa de extracción, en cada trama de la secuencia, de los rasgos que se quieren poner en correspondencia. De antemano se supone que dichos elementos son altamente identificables. Una vez hecho esto, la introducción

de modelos de movimiento lleva al planteamiento de un sistema de ecuaciones, normalmente no lineales. La solución puede plantearse entonces en dos etapas: en la primera, se establece la correspondencia de dichos rasgos entre tramas sucesivas; en la segunda, el desplazamiento observado para estos descriptores se usa para resolver el sistema de ecuaciones, lo cual finalmente lleva a la obtención de los parámetros de movimiento correspondientes a los objetos de la escena. Existen de esta manera soluciones basándose en correspondencia entre puntos, entre líneas rectas y entre líneas curvas [HILD87], [AGGA88], [HUAN92]. Las aplicaciones de estos métodos incluyen funciones de visión en robots, navegación, y control de tráfico, así como compresión de video orientada a objetos [AIZA95], [TEKA95].

### 2.3.2.2. Métodos de estimación por transformadas.

La motivación para estos métodos es el hecho de que determinadas características de una secuencia pueden resultar evidentes en el dominio de una transformación (transformada de Fourier, descomposición de Gabor, distribuciones de Wigner-Ville). Por ejemplo, un movimiento de translación puro representa un desfaseamiento en el dominio de la frecuencia. Por tanto, la medición de este cambio local de fase permite obtener los parámetros de translación buscados. Los métodos de correlación de fase (que son a la vez métodos basados en bloques) son el ejemplo más claro de este enfoque.

### 2.3.2.3. Métodos diferenciales.

Este enfoque es el utilizado en este trabajo. De manera general, estos métodos utilizan como hipótesis básica la *invariancia* de la luminancia o intensidad de las imágenes a lo largo de toda la secuencia. Esta hipótesis, llamada *restricción de flujo óptico*, se expresa matemáticamente como:

$$\frac{dI(\mathbf{z}, t)}{dt} = 0 \quad (2.9)$$

De esta forma, todas las variaciones en la intensidad se deben atribuir al movimiento. Así, si se supone un movimiento translacional puro, y no se consideran a las regiones de recubrimiento y descubrimiento, la ecuación (2.3) es válida si se conoce el *campo vectorial denso* de desplazamiento (un vector de movimiento por cada pixel). Ahora bien, este campo puede calcularse a partir (a) de la información local obtenida al derivar la función de intensidad de la secuencia con respecto al espacio y al tiempo, y (b) de restricciones adicionales impuestas para definir la estructura del movimiento. La hipótesis original de invariancia de la intensidad, junto con las suposiciones adicionales de *continuidad* y *diferenciabilidad* de  $I(\mathbf{z}, t)$  conduce a una ecuación diferencial, conocida como *ecuación de flujo óptico (EFO)* o *ecuación de*

restricción de movimiento aparente, que relaciona las derivadas espacio-temporales de la luminancia con el vector de velocidad  $\mathbf{v}$ :

$$\frac{\partial I(\mathbf{z}, t)}{\partial x} v_x(\mathbf{z}, t) + \frac{\partial I(\mathbf{z}, t)}{\partial y} v_y(\mathbf{z}, t) + \frac{\partial I(\mathbf{z}, t)}{\partial t} = \nabla I(\mathbf{z}, t) \bullet \mathbf{v} + \frac{\partial I(\mathbf{z}, t)}{\partial t} = 0 \quad (2.10)$$

Se hace notar que los vectores de velocidad y desplazamiento están relacionados por  $\mathbf{v} = d\mathbf{d}/dt$ . En el caso de una secuencia de imágenes digitales, si se supone que la velocidad se mantiene constante durante cada intervalo de tiempo  $\Delta t$ , se tiene que

$$d_x = v_x \Delta t \quad y \quad d_y = v_y \Delta t \quad (2.11)$$

por lo cual la estimación del flujo óptico y de los vectores de correspondencia son problemas equivalentes [BAAZ91b].

Los métodos diferenciales pueden dividirse en dos grandes grupos: los que utilizan *explícitamente* la ecuación de flujo óptico, y los métodos pel-recursivos, que también la emplean pero de manera *implícita*.

### Métodos basados en la ecuación de restricción de movimiento.

La EFO es insuficiente para determinar de manera única las dos componentes del campo de movimiento [HORN86]; con ella sólo puede obtenerse la componente del campo  $\mathbf{v}$  en la dirección paralela al gradiente de la intensidad (en la dirección normal a los contornos). En la dirección paralela a los contornos, el producto punto de la EFO es nulo, y no se puede calcular nada. Como se dijo, este resultado se conoce como problema de apertura.

Dada esta situación, para encontrar el campo de movimiento se añaden restricciones acerca de su estructura [BARR94]; estas hipótesis adicionales son generalmente de *suavizado* espacio-temporal, y han dado lugar a un número de métodos diferentes:

1. Horn y Schunck imponen una condición de *suavizado global máximo* en el campo de movimiento, esto es, una variación mínima entre los vectores de velocidad de pixel a pixel [HORN86]. El empleo conjunto de la EFO y de esta restricción global lleva a minimizar una función de costo de dos términos: uno relacionado con el ajuste del campo de movimiento a las variaciones espacio-temporales de la secuencia, y otro destinado a favorecer la continuidad del campo de desplazamiento.

2. Nagel y Enkelmann utilizan información de las segundas derivadas de la intensidad de la imagen, y sugieren una restricción de *suavizado adaptativo*: éste se suprime cuando se presentan gradientes de intensidad fuertes, y se refuerza en las direcciones donde hay variaciones pequeñas en los niveles de gris [NAGE86].
3. Lucas y Kanade suponen que el movimiento es constante por bloques de pixels [BARR94].

Existen además enfoques que utilizan otras restricciones de suavizado del campo [BARR94]. En general, los métodos que introducen suavizado global producen estimaciones imprecisas en las fronteras de movimiento, mientras que la introducción de suavizado adaptativo (que representa consideraciones locales explícitas) permite la presencia de discontinuidades en el campo [TEKA95].

### **Métodos pel-recursivos.**

Se desarrollaron originalmente para codificación de señales de televisión [NETR79]. Estos métodos estiman un campo denso de vectores de desplazamiento a través de una *actualización recursiva y/o iterativa* del vector de desplazamiento en cada pixel de la secuencia, basada en la minimización del error de reconstrucción de la imagen predicha. Se considera que tales algoritmos adoptan la ecuación de restricción de movimiento de manera implícita, ya que son estimadores del tipo *predictor-corrector*: una predicción inicial del desplazamiento correspondiente a cada pixel se actualiza o corrige de acuerdo a la información de los gradientes espacio-temporales existentes en cada punto de la imagen, y al criterio de optimización. La predicción inicial del desplazamiento se elige como la estimación final realizada para algún otro pixel previamente procesado. Los algoritmos pel-recursivos fueron desarrollados ampliamente durante los años ochenta [NETR80b], [HUAN81], [HUAN83], [PAQU83], [SABR84], [WALK84], [BIEM87], [MOOR87], [WALK87], [TZIR88], [BAAZ91b], y se describirán con detalle en los dos capítulos siguientes.

#### **2.3.2.4. Métodos bayesianos.**

Pueden verse como una extensión general de los métodos diferenciales. La *modelización markoviana* facilita la introducción de distintas fuentes de información dentro de un solo esquema, así como la especificación de interacciones locales complejas entre información de intensidad de la imagen, vectores y discontinuidades de movimiento, y regiones de oclusión, todo dentro de una función de costo global a optimizar (criterio *máximo a posteriori*). Estos métodos utilizan restricciones de suavizado del campo de desplazamiento de tipo probabilístico, bajo la forma de un campo aleatorio de Gibbs, que favorece la asignación de una probabilidad a priori más alta a campos vectoriales más suaves. Para incluir restricciones de suavizado direccionales, se introducen modelos más complejos que consideran, por ejemplo, campos de líneas de discontinuidad de movimiento [KONR92], [HEIT93], [BRAI95a], [TEKA95].

El enfoque bayesiano se describirá con más detalle posteriormente, pero para su aplicación en segmentación basada en movimiento (capítulos 6 y 7).

## **2.4 Conclusiones.**

Se presentaron en este capítulo distintos aspectos del problema de la estimación de movimiento aparente en una secuencia de imágenes, así como una descripción general de los numerosos métodos desarrollados para abordarlo. Debe quedar claro que:

- cada enfoque de estimación tiene sus puntos fuertes y débiles.
- la calidad de una técnica depende de la naturaleza de los procesos que harán uso posterior o interpretarán dichas estimaciones.
- si bien el problema básico de estimación ha sido resuelto, quedan abiertas aún muchas rutas para optimizar los algoritmos existentes, desarrollar técnicas novedosas, y aplicarlos a nuevos problemas, cada vez mas complejos.

Ya que el éxito de los procesos subsecuentes al de estimación (procesamiento posterior o aplicaciones) depende de las propiedades del campo de movimiento obtenido, es muy importante conocer las características particulares de las técnicas de estimación de movimiento aparente. En el capítulo siguiente, se presentará con más profundidad el enfoque de estimación utilizado en este trabajo, que es el de los métodos pel-recursivos.

## Capítulo 3

# Los algoritmos pel-recursivos de estimación de movimiento.

### 3.1 Introducción.

Como se mencionó, todos los métodos diferenciales utilizan la EFO acompañada de restricciones de suavizado. Los métodos pel-recursivos la usan de manera implícita, al ser estimadores del tipo predicción-corrección.

Estos algoritmos se desarrollaron para codificación de señales de televisión con movimientos lentos, bajo un esquema *DPCM con compensación de movimiento* (DPCM-MC) [GATI96]. Buscaron ser fáciles de implantar en circuitería, por lo cual su lógica de operación original era bastante simple y su desempeño en cuanto a la estimación del movimiento, más bien pobre. Posteriormente, se desarrollaron enfoques más complejos que permitieron mejorar tanto la estimación del movimiento como la reconstrucción de las secuencias comprimidas. En este capítulo se presentan la descripción y evolución de estos algoritmos.

#### 3.1.1 La diferencia de trama desplazada (DFD).

Los primeros métodos de codificación por compensación de movimiento (pel-recursivos y de puesta en correspondencia entre bloques) basaron su funcionamiento en las siguientes suposiciones [MUSM85]: iluminación de escena constante (ecuación 2.9), movimiento translacional puro, e inexistencia de regiones de oclusión. En este caso idealizado, es válido emplear la ecuación (2.3), que relaciona pares de pixels en tramas sucesivas, según el vector de desplazamiento que los conecte

$$I(\mathbf{z}, t) = I(\mathbf{z} - \mathbf{d}(x, y, t), t - k\Delta t)$$

Se define una cantidad, conocida como *diferencia de trama desplazada* (DFD por sus iniciales en inglés), como

$$DFD(\mathbf{z}, \mathbf{d}(x, y, t)) = I(\mathbf{z}, t) - I(\mathbf{z} - \mathbf{d}(x, y, t), t - k\Delta t) \quad (3.1)$$

que depende de dos variables: la posición  $\mathbf{z}$  y el desplazamiento  $\mathbf{d}$  correspondiente a la posición  $\mathbf{z}$ . La  $DFD$  indica qué tan bien el valor del desplazamiento  $\mathbf{d}$  permite reconstruir cada pixel de la trama actual  $I(\mathbf{z}, t)$  a partir de la información existente en la trama anterior  $I(\mathbf{z}, t - k\Delta t)$ . Por tanto, puede definirse como un error de reconstrucción.

La minimización del error de reconstrucción, junto con la cantidad de información a transmitir, y la complejidad del sistema transmisor-receptor, son los tres criterios que rigen el desempeño de un sistema de codificación de imágenes. De esta forma, en un sistema que utilice información de movimiento para codificar secuencias de imágenes, la  $DFD$  aparece siempre como un parámetro importante.

En la práctica, la  $DFD$  difícilmente es cero, porque no se satisfacen las hipótesis originales, y debido a otras razones, como al ruido presente en la secuencia. Además, es pertinente observar que  $\mathbf{d}$  puede ser un vector con componentes reales, y por tanto, el cálculo de la  $DFD$  implica el uso de una operación de interpolación en la imagen de la trama anterior, que también introduce errores numéricos.

### 3.2 El algoritmo pel-recursivo básico.

Los algoritmos de codificación de imágenes por estimación y compensación de movimiento surgieron como alternativa a los codificadores de rellenado condicional (*conditional replenishment*) desarrollados hacia fines de los años sesenta [JAIN89]. Las primeras propuestas de técnicas de estimación de movimiento para procesamiento de señales de televisión datan de mediados de los años setenta [LIMB75], [CAFF76].

En 1979, Netravali y Robbins propusieron el primer algoritmo pel-recursivo para la estimación del campo de desplazamiento  $\hat{\mathbf{d}}$  [NETR79], en el cual a partir de una estimación inicial del desplazamiento  $\hat{\mathbf{d}}^{i-1}$  (predicción) se obtiene una estimación mejor  $\hat{\mathbf{d}}^i$ :

$$\hat{\mathbf{d}}^i = \hat{\mathbf{d}}^{i-1} + \mathbf{u} \quad (3.2)$$

donde  $\mathbf{u}$  es el término de actualización (corrección) de la  $i$ -ésima recursión. La etapa de predicción impone una restricción de suavizado local en la estimación, ya que originalmente, la estimación final  $\hat{\mathbf{d}}^i$  calculada para un pixel se convierte en la predicción  $\hat{\mathbf{d}}^{i-1}$  para el siguiente pixel a procesar.

Esta técnica se basa en la minimización recursiva de  $DFD^2$ , actualizando la estimación del vector de desplazamiento en cada pixel. Así, dada la  $i$ -ésima estimación del desplazamiento, se obtiene la  $(i+1)$ -ésima estimación tal que, en

general, el error de reconstrucción resultante de esta estimación sea cada vez más pequeño:

$$\hat{\mathbf{d}} = \operatorname{argmin}_{\mathbf{d}}(DFD^2(\mathbf{z}, \mathbf{d})) \quad (3.3)$$

La minimización se realiza por el método de *máximo descenso* (*steepest descent*), que es un método clásico de minimización de funciones - y quizás también el más simple - que establece en esencia que el negativo del gradiente de una función apunta en la dirección en la cual la función disminuye más rápidamente [BEAL88] La figura 3.1 ilustra este método.

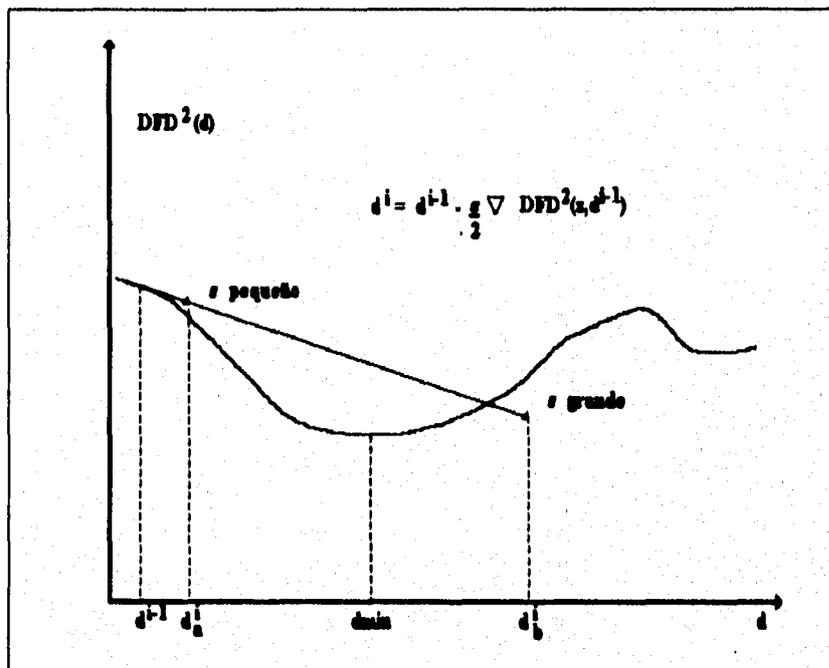


Figura 3.1: Método de máximo descenso (steepest descent)

La minimización es *recursiva* a través de la imagen (pixel a pixel, siguiendo la dirección del barrido de una señal de televisión: de izquierda a derecha y de arriba hacia abajo) y puede ser *iterativa* (iteraciones múltiples para un mismo pixel). De aquí proviene el nombre de *pel-recursivo*.

La  $DFD$  debe converger a cero cuando la estimación  $\hat{\mathbf{d}}$  converja al desplazamiento "real" de la imagen  $\mathbf{d}$  (hay que recordar que el método de máximo descenso es un método de *optimización local*; por lo tanto, sólo se asegura encontrar un  $\hat{\mathbf{d}}^i$  que corresponda a un mínimo local de la  $DFD^2$ ). Como ya se dijo, la  $DFD$  no es cero normalmente, por lo que no se alcanza una solución exacta. En cambio, se introduce

un umbral  $T_{DFD}$  preestablecido, de forma tal que si la  $DFD$  correspondiente a una estimación  $\hat{\mathbf{d}}^i$  cumple que  $|DFD(\mathbf{z}, \hat{\mathbf{d}}^i)| < T_{DFD}$ , entonces la solución se considera aceptable [BAAZ91b].

La expresión recursiva obtenida por el método de máximo descenso es:

$$\hat{\mathbf{d}}^i = \hat{\mathbf{d}}^{i-1} - \frac{\epsilon}{2} \nabla_d DFD^2(\mathbf{z}, \hat{\mathbf{d}}^{i-1}) \quad (3.4)$$

donde

- $\hat{\mathbf{d}}^i$  es la estimación del vector de desplazamiento para la  $i$ -ésima iteración para el pixel presente.
- $\hat{\mathbf{d}}^{i-1}$  es la estimación inicial (predicción) del vector de desplazamiento.
- $\epsilon$  es un coeficiente de convergencia fijo.
- $\nabla_d$  es el operador gradiente con respecto al desplazamiento.

Al desarrollar el gradiente  $\nabla_d$  se obtiene

$$\hat{\mathbf{d}}^i = \hat{\mathbf{d}}^{i-1} - \epsilon DFD(\mathbf{z}, \hat{\mathbf{d}}^{i-1}) \nabla I(\mathbf{z} - \hat{\mathbf{d}}^{i-1}, t - k\Delta t) \quad (3.5)$$

donde ahora  $\nabla I$  es el gradiente de la intensidad de la imagen.

Esta fórmula indica que en cada iteración se le agrega a la estimación anterior una cantidad vectorial paralela a la dirección del gradiente espacial de la intensidad de la imagen (recordar el problema de apertura), y cuya magnitud es proporcional al error de compensación de movimiento. Se hace énfasis en que  $\hat{\mathbf{d}}^i$  representa el valor actual del desplazamiento. En cambio,  $\hat{\mathbf{d}}^{i-1}$  puede ser la última estimación realizada para el pixel anterior (en la dirección de barrido), la estimación más reciente calculada para el pixel actual (en el caso de múltiples iteraciones por pixel), o algún otro valor. En el algoritmo original, siempre que se calcula un nuevo  $\hat{\mathbf{d}}^i$ , el  $\hat{\mathbf{d}}^i$  viejo se convierte en  $\hat{\mathbf{d}}^{i-1}$ , y así sucesivamente.

La presencia de bordes con distintas orientaciones en una escena es lo que permite la convergencia de  $\hat{\mathbf{d}}^i$  a un  $\mathbf{d}$  que satisfaga el criterio de minimización de  $DFD^2$ . El método funciona porque, bajo condiciones ideales, una vez obtenido un vector de desplazamiento "correcto", éste no debe variar drásticamente de un pixel a otro si ambos forman parte de una región con el mismo movimiento. Los cambios considerables en el desplazamiento ocurren regularmente en las fronteras de movimiento. En estas regiones son necesarios nuevos períodos de convergencia.

El valor del coeficiente de convergencia  $\epsilon$  se elige de manera heurística y, como en cualquier problema resuelto por el método de máximo descenso, establece un compromiso entre la velocidad y la precisión en la convergencia (figura 3.1):

- valores de  $\epsilon$  grandes producen una convergencia rápida pero una estimación poco precisa, o bien oscilaciones alrededor del mínimo local hallado.
- valores de  $\epsilon$  pequeños permiten más precisión en la estimación, pero al costo de un número mayor de iteraciones.

Posteriormente, en [MOOR87] se realiza un análisis para determinar las condiciones necesarias para la convergencia en la estimación del desplazamiento en este algoritmo.

Una versión simplificada que facilita la implantación en circuitería es la siguiente:

$$\hat{d}^i = \hat{d}^{i-1} - \epsilon \operatorname{sign}(DFD(\mathbf{z}, \hat{d}^{i-1})) \operatorname{sign}(\nabla I(\mathbf{z} - \hat{d}^{i-1}, t - k\Delta t)) \quad (3.6)$$

donde  $\operatorname{sign}()$  representa la función  $\operatorname{signo}(x)$ .

El algoritmo original puede extenderse al considerar una vecindad espacial para imponer una restricción de suavizado local más fuerte en la estimación de  $\hat{d}^i$ , a costa de un incremento en el costo computacional [NETR80b]. En esta extensión,  $\hat{d}^i$  se actualiza usando el método de máximo descenso para minimizar una suma ponderada de la  $DFD$  sobre una vecindad espacial causal:

$$\hat{d}^i = \hat{d}^{i-1} - \epsilon \sum_{n \in N_{vec}} W_j DFD(\mathbf{z}_n, \hat{d}^{i-1}) \nabla I(\mathbf{z}_n - \hat{d}^{i-1}, t - k\Delta t) \quad (3.7)$$

donde  $W_j$  son los factores de peso, y cumplen con las propiedades de una función de probabilidad.

En todos los algoritmos anteriores el número de iteraciones que deben realizarse por cada pixel para obtener un vector de desplazamiento que cumpla con el criterio de minimización es variable. Además, existen casos en los cuales no se obtiene la convergencia aún después de una cantidad grande de iteraciones, por lo que siempre se fija un tope máximo en el número de iteraciones por pixel.

### 3.3 El algoritmo de Walker y Rao.

En 1984, en [WALK84] se propone una modificación al algoritmo original, que mejora notablemente la velocidad de convergencia. En el algoritmo de [NETR79], puede observarse que:

- Las correcciones más grandes en la estimación del vector  $\hat{\mathbf{d}}^i$  deben ocurrir cuando  $|DFD|$  sea grande y  $|\nabla I|$  sea pequeña (como en una región uniforme), porque el error de reconstrucción es considerable, y por tanto, hay que actualizar adecuadamente el vector de desplazamiento.
- En el caso de un borde ( $|\nabla I|$  grande),  $\epsilon$  debe ser pequeño si la magnitud de la  $|DFD|$  es pequeña, para que la actualización no sea excesiva.

Lo anterior implica que  $\epsilon$  debe mantenerse pequeño para permitir estas correcciones mínimas; pero un valor pequeño para  $\epsilon$  implica un gran número de iteraciones cuando se requieren correcciones grandes. Ante este compromiso, Walker y Rao hacen que  $\epsilon$  sea un factor variable, inversamente proporcional al gradiente de la intensidad. De esta forma:

$$\hat{\mathbf{d}}^i = \hat{\mathbf{d}}^{i-1} - \epsilon' DFD(\mathbf{z}, \hat{\mathbf{d}}^{i-1}) \nabla I(\mathbf{z} - \hat{\mathbf{d}}^{i-1}, t - k\Delta t) \quad (3.8)$$

donde

$$\epsilon' = \frac{1}{2|\nabla I(\mathbf{z} - \hat{\mathbf{d}}^{i-1}, t - k\Delta t)|^2} \quad (3.9)$$

Este término adaptativo se propuso originalmente de manera heurística, pero se puede hallar también analíticamente, utilizando la EFO [BAAZ91b]. Se encontró en la práctica que el término adaptativo mejora en mucho el desempeño del algoritmo en cuanto a velocidad de convergencia.

### 3.4 El algoritmo basado en estimación de Wiener.

Este algoritmo [BIEM87] proporciona una estimación lineal por mínimos cuadrados del término de actualización del vector de desplazamiento

$$\mathbf{u} = \mathbf{d} - \hat{\mathbf{d}}^{i-1} \quad (3.10)$$

utilizando  $N$  observaciones para obtener una estimación más confiable del vector de movimiento. Aquí,  $\mathbf{d}$  representa el desplazamiento real de cada pixel, y  $\hat{\mathbf{d}}^{i-1}$  la  $(i-1)$ -ésima estimación de éste. El método supone que el término de actualización y el error de linealización de la  $DFD$  son muestras de procesos estocásticos.

A partir de la definición para la  $DFD$ , se puede escribir:

$$DFD(\mathbf{z}, \hat{\mathbf{d}}^{i-1}) = I(\mathbf{z} - \mathbf{d}, t - k\Delta t) - I(\mathbf{z} - \hat{\mathbf{d}}^{i-1}, t - k\Delta t) \quad (3.11)$$

La linealización de la  $DFD$  mediante la expansión por series de Taylor de la función de intensidad produce

$$DFD(\mathbf{z}, \hat{\mathbf{d}}^{l-1}) = -\nabla^T I(\mathbf{z} - \hat{\mathbf{d}}^{l-1}, t - k\Delta t) \cdot (\mathbf{d} - \hat{\mathbf{d}}^{l-1}) + v(\mathbf{z}, \hat{\mathbf{d}}^{l-1}) \quad (3.12)$$

El término de error  $v(\mathbf{z}, \hat{\mathbf{d}}^{l-1})$ , correspondiente a los términos de orden superior a uno de la linealización de la  $DFD$ , no se desprecia sino que se considera como una muestra de un proceso estocástico (ruido). Asimismo, el vector  $(\mathbf{d} - \hat{\mathbf{d}}^{l-1})$  se considera salida de un proceso estocástico.

Las  $N$  observaciones representan  $N$  puntos  $\mathbf{z}_j$ ,  $j = 1, \dots, N$  en alguna vecindad de cada pixel analizado. Si este pixel pertenece a alguna región en movimiento, entonces todos aquellos pixels pertenecientes a la vecindad preestablecida se usan para encontrar una estimación del desplazamiento más estable. En el algoritmo original, la vecindad es espacial, pero en el caso más general puede ser espacio temporal o incluso espectral (ver sección 3.6). Así, aplicando la ecuación anterior a cada uno de las  $N$  observaciones:

$$\begin{pmatrix} DFD(\mathbf{z}_1, \hat{\mathbf{d}}^{l-1}) \\ \vdots \\ DFD(\mathbf{z}_N, \hat{\mathbf{d}}^{l-1}) \end{pmatrix} = - \begin{pmatrix} I_x^1 & I_y^1 \\ \vdots & \vdots \\ I_x^N & I_y^N \end{pmatrix} \cdot (\mathbf{d} - \hat{\mathbf{d}}^{l-1}) + \begin{pmatrix} v(\mathbf{z}_1, \hat{\mathbf{d}}^{l-1}) \\ \vdots \\ v(\mathbf{z}_N, \hat{\mathbf{d}}^{l-1}) \end{pmatrix} \quad (3.13)$$

donde

$$I_x^j = I_x(\mathbf{z}_j - \hat{\mathbf{d}}^{l-1}, t - k\Delta t)$$

$$I_y^j = I_y(\mathbf{z}_j - \hat{\mathbf{d}}^{l-1}, t - k\Delta t)$$

son las componentes del gradiente  $\nabla I$  en cada pixel  $\mathbf{z}_j$  desplazado. Esta ecuación matricial puede por tanto escribirse como

$$\mathbf{e} = \mathbf{G}\mathbf{u} + \mathbf{v} \quad (3.14)$$

donde

$\mathbf{e}$  es un vector columna de  $N \times 1$  formado por las  $DFD(\mathbf{z}_j, \hat{\mathbf{d}}^{l-1})$ .

$\mathbf{u} = \mathbf{d} - \hat{\mathbf{d}}^{i-1}$  es un vector columna de  $2 \times 1$  (término de actualización).

$\mathbf{v}$  es un vector columna de  $N \times 1$  errores de truncatura  $v(\mathbf{z}, \hat{\mathbf{d}}^{i-1})$ .

$G$  es la matriz de  $N \times 2$  de gradientes .

Con el modelo anterior se pretende hallar una estimación lineal de mínimos cuadrados  $\hat{\mathbf{u}}$  de  $\mathbf{u}$  dadas  $\mathbf{e}$  y  $G$ . El problema de estimación consiste en encontrar un estimador lineal  $L$

$$\hat{\mathbf{u}} = L\mathbf{e} \quad (3.15)$$

tal que  $E\{\|\mathbf{u} - \hat{\mathbf{u}}\|^2\}$  se minimice sujeta a la ecuación anterior. Este estimador es el conocido *estimador de Wiener* [HAYK89].

El desarrollo matemático para la obtención del estimador lineal  $L$ , bajo la suposición de ortogonalidad entre el ruido  $\mathbf{v}$  y el término de actualización  $\mathbf{u}$  (inter-correlación nula) lleva a obtener una expresión para la estimación de  $\hat{\mathbf{u}}$  (el desarrollo en detalle puede verse en [BIEM87]):

$$\hat{\mathbf{u}} = [G^T R_v^{-1} G + R_u^{-1}]^{-1} G^T R_v^{-1} \cdot \mathbf{e} \quad (3.16)$$

donde  $R_u$  y  $R_v$  son las matrices de covarianza de  $\mathbf{u}$  y  $\mathbf{v}$  respectivamente. De esta forma, la expresión para la actualización del vector de desplazamiento  $\hat{\mathbf{d}}$  se encuentra por la expresión típica de los algoritmos pel-recursivos:

$$\hat{\mathbf{d}} = \hat{\mathbf{d}}^{i-1} + \hat{\mathbf{u}} \quad (3.17)$$

La actualización se ejecuta de manera similar a los métodos anteriores (recursivamente a lo largo de la línea de barrido).

La solución anterior requiere del conocimiento de las matrices de covarianza del término de actualización y del error de linealización  $R_u$  y  $R_v$ . Para simplificar la expresión anterior, se introducen las hipótesis de que ambas matrices tienen media nula y que están mutuamente no correlacionadas, esto es  $R_u = \sigma_u^2 \mathbf{I}_2$  y  $R_v = \sigma_v^2 \mathbf{I}_N$  donde  $\sigma_u^2$  y  $\sigma_v^2$  representan las variancias del término de actualización y del error de linealización, respectivamente, e  $\mathbf{I}_m$  representa la matriz identidad de tamaño  $m$ . Con ésto se llega a una expresión más sencilla para el cálculo de la actualización recursiva del desplazamiento  $\hat{\mathbf{d}} = \hat{\mathbf{d}}^i$ :

$$\hat{d}^l = \hat{d}^{l-1} - \begin{pmatrix} \sum_{j=1}^N (I_x^j)^2 + \mu & \sum_{j=1}^N I_x^j I_y^j \\ \sum_{j=1}^N I_x^j I_y^j & \sum_{j=1}^N (I_y^j)^2 + \mu \end{pmatrix}^{-1} \begin{pmatrix} \sum_{j=1}^N I_x^j \cdot DFD(z_j, \hat{d}^{l-1}) \\ \sum_{j=1}^N I_y^j \cdot DFD(z_j, \hat{d}^{l-1}) \end{pmatrix} \quad (3.18)$$

donde se introduce un término de amortiguamiento  $\mu = \sigma_v^2 / \sigma_u^2$ . A continuación se muestran algunas configuraciones posibles para la vecindad  $N$ .

X X	X X X	X X X X	X X X X X	X : pixel vecino
X O	X O	X X O	X X X O	O : pixel actual
$N = 4$	$N = 5$	$N = 7$	$N = 9$	

El análisis de la forma final del algoritmo muestra que de él se pueden derivar algunos otros algoritmos pel-recursivos:

1. Si  $N=1$ ,  $\mu = 100$ , se obtiene el algoritmo simplificado de Cafforio y Rocca [CAFF83].
2. Si  $N=1$ ,  $\mu = 0$ , y se multiplica el término de actualización por  $\frac{1}{2}$ , se obtiene el algoritmo de Walker y Rao.

### 3.5 Lógica de los algoritmos pel-recursivos de estimación de movimiento.

Las ecuaciones (3.5), (3.7), (3.8) y (3.18) permiten estimar el vector de desplazamiento pixel a pixel de la secuencia de imágenes. Pero se encontró en [WALK84] que era conveniente añadir una serie de condiciones a estas ecuaciones para que los procesos de estimación de movimiento y compresión de datos mejoraran.

Se introduce primero el concepto de *pixel compensado* como aquél para el cual se obtiene una estimación del desplazamiento  $\hat{d}$  que cumpla la condición de convergencia ( $|DFD(z, \hat{d})| < T_{DFD}$ , donde  $T_{DFD}$  es el umbral de convergencia).

Pueden plantarse varios algoritmos [NETR79], [WALK84], [BIEM87], [TZIR87], [BAAZ91a], pero en todos se incluyen los siguientes aspectos:

1. Selección de un valor inicial para la estimación del vector de desplazamiento ( $\hat{d}^{l-1} = \hat{d}^0$ ).

2. *Prueba de detección de movimiento.* Tiene como finalidad asociar un desplazamiento nulo a los pixels inmóviles. La complejidad del algoritmo de detección de áreas fijas y móviles varía desde un simple proceso de comparación de la *diferencia interimagen o temporal FD*

$$FD(\mathbf{z}, t) = I(\mathbf{z}, t) - I(\mathbf{z}, t - k\Delta t) \quad (3.19)$$

con un umbral  $T_{FD}$ , hasta técnicas sofisticadas [BIER86], [HOTT88]. Para aplicaciones en codificación de tipo DPCM-MC, sólo se dispone de la información previamente transmitida al receptor, lo que limita la elección de uno de estos algoritmos. Por esa razón, típicamente se calcula la *FD* puntualmente o en una vecindad causal para cada pixel, y se compara con  $T_{FD}$ . Si  $|FD(\mathbf{z}, t)| < T_{FD}$ ,  $\hat{\mathbf{d}}^i = \mathbf{0}$ , y el pixel en cuestión es compensado.

3. *Prueba de recursividad.* Si para el desplazamiento inicial  $\hat{\mathbf{d}}^{i-1}$ , se cumple que  $|DFD(\mathbf{z}, \hat{\mathbf{d}}_{i-1}^i)| < T_{DFD}$ , se concluye que dicha estimación es suficientemente buena, y por tanto, no se necesita actualizar:  $\hat{\mathbf{d}}^i = \hat{\mathbf{d}}^{i-1}$ . Esto representa una situación de *inercia* en el movimiento estimado, que puede verse también como una condición de suavizado adicional. El pixel en cuestión es compensado.
4. *Prueba para restablecimiento de  $\hat{\mathbf{d}}^{i-1}$ .* Combinando los criterios anteriores, puede encontrarse que la estimación inicial sea muy mala, y que conviene recomenzar el proceso de estimación con un vector nulo. Esto puede incidir en un número mayor de iteraciones, pero con mejor convergencia.
5. *Prueba de iteratividad.* Si las primeras dos pruebas no se cumplen, debe emplearse el algoritmo pel-recursivo para actualizar el vector de desplazamiento, hasta que (a) se cumpla la condición de convergencia, o (b) se llegue al tope máximo en el número de iteraciones  $it_{max}$ .
6. Finalmente, habrán pixels para los cuales no se encuentre la convergencia después de todas las pruebas anteriores. A estos pixels se les llama *pixels no compensados*.

Los algoritmos de las secciones anteriores pueden ser utilizados con una misma lógica general. En este trabajo, se elige la lógica usada en [WALK84] y [BAAZ91b].

### 3.6 Compensación de movimiento.

Una vez estimado el campo denso de desplazamiento, cada trama  $I(\mathbf{z}, t)$  de la secuencia se reconstruye a partir de la trama anterior  $I(\mathbf{z}, t - 1)$  ( $k\Delta t = 1$ ) y del campo estimado  $\hat{\mathbf{d}}$ , siguiendo la dirección de barrido. Para cada  $\mathbf{z}$ :

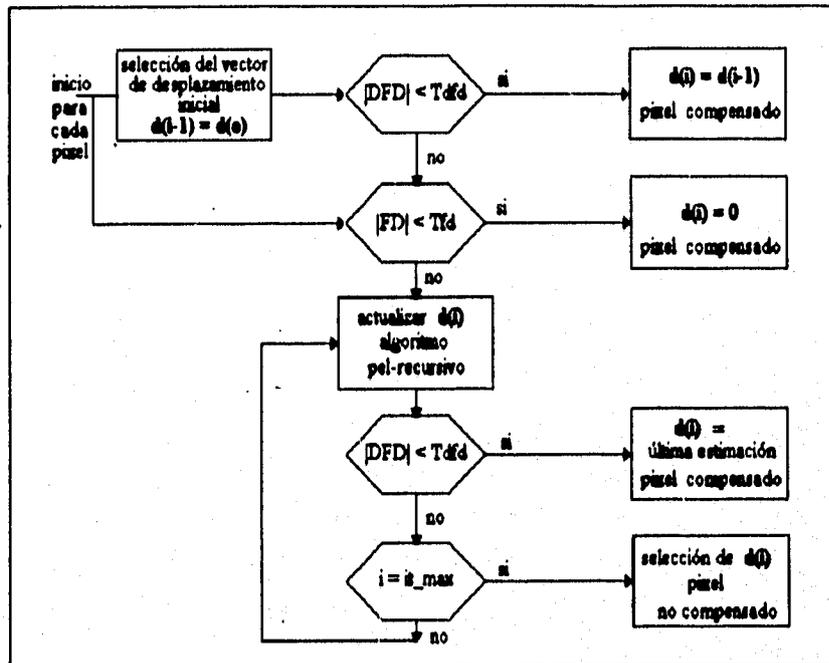


Figura 3.2: Lógica de los algoritmos pel-recursivos.

$$\hat{I}(z, t) = I(z - \hat{d}, t - 1) \quad (3.20)$$

### 3.7 Variaciones de los algoritmos pel-recursivos.

#### 3.7.1 Algoritmos combinados WR-BD.

En [BAAZ91a] se realizan las siguientes observaciones:

- El método de Walker y Rao (WR) es un caso particular del método de Biemond et al. (BD).
- Las operaciones aritméticas en ambos casos son las mismas.
- Las ecuaciones pueden usarse con una misma lógica de funcionamiento.
- En el algoritmo BD se supone que las  $N$  observaciones (pixels) pertenecen a una región con el mismo movimiento. Por tanto, se espera que funcione bien cuando se cumple esta hipótesis, y que se introduzcan errores en caso contrario (como en las fronteras de movimiento).

- Esta característica de suavizado del campo no ocurre en WR, donde la estimación es puntual.

Con base en esto, Baaziz propone combinar ambos métodos, como una forma sencilla de introducir una vecindad adaptativa simplificada, y conmutar entre una estimación con una vecindad de  $N$  puntos, y una estimación puntual. Ahora bien, el criterio de conmutación entre uno y otro método es heurístico, ya que no se realizan pruebas elaboradas para conocer el tamaño de vecindad requerido. El criterio que rige la utilización de WR o BZ en cada pixel es nuevamente la minimización de la *DFD*.

Es claro que la solución ideal es la introducción de un detector de *fronteras o discontinuidades de movimiento* que adapte el tamaño de la vecindad sobre la cual se hace la estimación.

Baaziz propone entonces dos variantes del método:

1. Para cada pixel, una vez elegido  $\hat{d}^0$ , se actualiza el vector de desplazamiento por BD y simultáneamente por WR. Con estos vectores  $\hat{d}_{BD}^i$  y  $\hat{d}_{WR}^i$ , se calcula la *DFD* para cada caso, y se elige como estimación final  $\hat{d}^i$  el vector que produzca una *DFD* menor [BAAZ91a].
2. Para cada pixel, una vez elegido  $\hat{d}^0$ , se aplica primero el método BD. Solamente en caso que el pixel analizado no sea compensado, se aplica el método WR [BAAZ91b].

Las figuras 3.3 y 3.4 describen los algoritmos descritos. En lo sucesivo, se hará referencia a los métodos 1 y 2 anteriores como BZ1 y BZ2, respectivamente.

### 3.7.2 Métodos a múltiples resoluciones (MR).

Los algoritmos pel-recursivos pueden aplicarse jerárquicamente utilizando una representación a múltiples resoluciones (pirámides gaussianas, laplacianas o transformada de ondículas (wavelets)) para obtener mejores resultados. La idea de estimar el movimiento jerárquicamente ha sido usada por numerosos autores [ANAN93], apoyada por los siguientes argumentos:

- Una estrategia jerárquica puede acelerar la convergencia de los algoritmos, y por tanto, disminuir el tiempo de cálculo.
- Estas técnicas permiten la estimación de desplazamientos de mayor magnitud, que con los algoritmos originales podrían no estimarse con precisión.

Varios estudios han ilustrado la eficiencia de esta estrategia, no solamente con los métodos pel-recursivos sino con otros (puesta en correspondencia de bloques, enfoques markovianos).

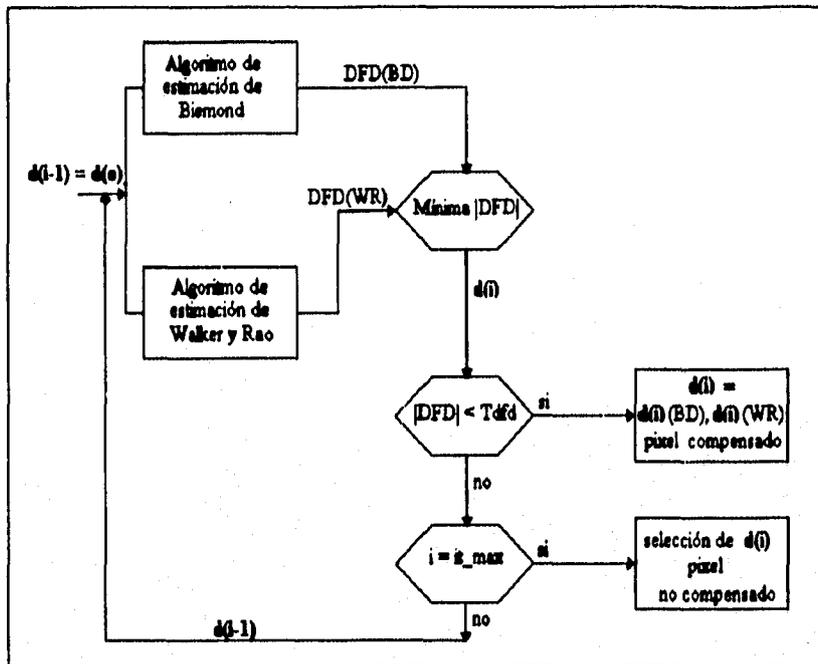


Figura 3.3: Algoritmo WR-BD combinado opción 1.

El concepto básico de la estimación MR con algoritmos pel-recursivos es la siguiente: el algoritmo inicia en el nivel más alto de la pirámide (resolución más baja). En este nivel se aplica el método pel-recursivo. El campo de movimiento obtenido sirve como estimación inicial para el siguiente nivel de resolución, y el procedimiento continúa hasta llegar a la resolución original. En los niveles más bajos de la pirámide, donde se distinguen las estructuras finas de la imagen, se realiza una estimación más precisa del desplazamiento, partiendo de una buena estimación inicial proveniente del nivel anterior. En [BAAZ91a] y [BAAZ91b] se aplican los métodos de WR, BD y BZ bajo esta perspectiva, utilizando pirámides gaussianas, reportándose mejoría en la calidad de la estimación (en cuanto a las características de coherencia del campo) y de la reconstrucción de las imágenes, pero también una complejidad computacional mayor (sumados los cálculos de la estimación de movimiento y de la descomposición piramidal) que la de los algoritmos a una sola resolución.

Una variante interesante es propuesta por Baaziz [BAAZ91b] al aplicar el método basado en Wiener a una descomposición espectral por transformada de ondículas, explotando las diferentes subbandas de frecuencia existentes (BB, BH, HB y HH) en una pirámide de ondículas. La suposición básica es que existe coherencia en el movimiento entre las diferentes subbandas de la imagen, es decir, que los pixels correspondientes a las diferentes componentes frecuenciales de un objeto en movimiento están relacionados con el mismo vector de desplazamiento. Por lo

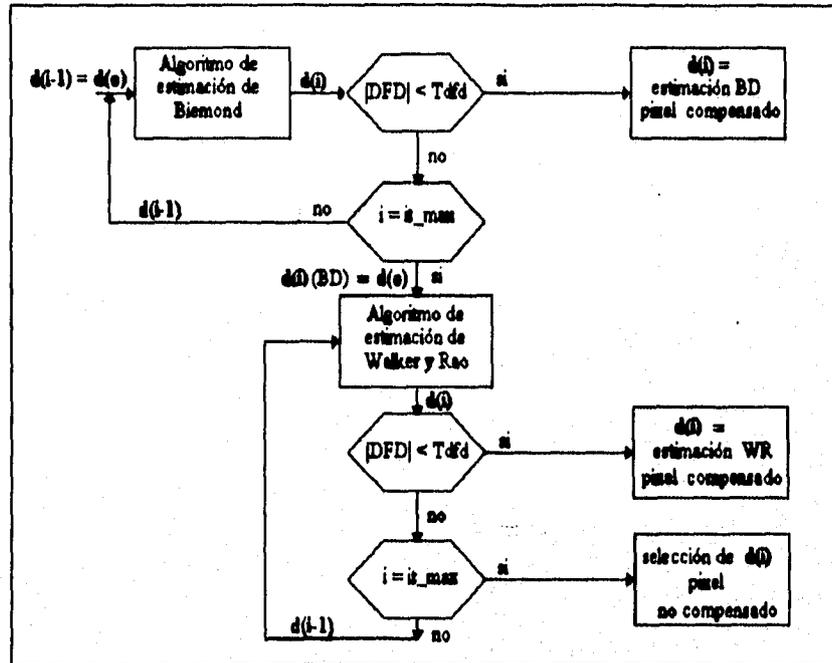


Figura 3.4: Algoritmo WR-BD combinado opción 2.

tanto, para el método basado en Wiener, la vecindad de las  $N$  observaciones se elige no espacialmente como en el caso original, sino *espectralmente* sobre las cuatro subbandas de la descomposición de la imagen original. Esto ocurre en cada nivel de la pirámide.

Este algoritmo tiene una lógica de funcionamiento similar a otros algoritmos a múltiples resoluciones. Los resultados muestran que este método, comparado con el caso a una sola resolución, tiene tres ventajas: se obtiene un porcentaje de píxeles sin compensar más bajo, tiene un costo computacional menor, y permite obtener una tasa de compresión más alta, para una muy buena calidad de las imágenes reconstruidas.

### 3.8 Conclusiones.

En este capítulo se presentaron los algoritmos pel-recursivos más representativos, así como algunas de las variantes que se han propuesto para mejorar su comportamiento. En el capítulo siguiente se muestran los resultados obtenidos al implantar estos algoritmos, junto con un análisis de su desempeño.

## Capítulo 4

# Implantación y evaluación de los algoritmos pel-recursivos.

### 4.1 Criterios de evaluación de los algoritmos.

¿Cuáles son las cantidades y propiedades que interesa evaluar en un estimador de movimiento? Estas dependen, entre otras cosas, de su aplicación particular. Sin embargo, se puede hablar de los siguientes aspectos generales:

**1. Calidad de la estimación.** Se ha visto hasta ahora que un estimador de movimiento pel-recursivo produce un campo denso de vectores de desplazamiento aparente. Evaluar la congruencia entre un campo de movimiento estimado y la realidad puede realizarse tanto cuantitativa como cualitativamente.

Dos cuestiones previas: en primer lugar, el campo de movimiento de una secuencia de imágenes real cualquiera es en general *desconocido* ya que existen fuentes y tipos de movimiento variados y complejos, y distintas cantidades de ruido. Se puede tener una idea general del campo, mas no los valores exactos de desplazamiento para cada pixel. En segundo término, se dijo en el capítulo 2 que este campo puede no corresponder exactamente al movimiento real de los objetos [HORN86], [HILD87], [VERR89], pero debe ser una buena aproximación.

En vista de lo anterior, el uso de secuencias de imágenes sintéticas representa una alternativa para la evaluación de la calidad del campo [BARR94]. En ellas pueden controlarse de manera sistemática tanto las características del campo de movimiento como las propiedades de la escena (ruido, condiciones de oclusión, iluminación variable). Como se conoce el valor real del desplazamiento, puede evaluarse cuantitativamente la precisión numérica de los métodos, obtener medidas estadísticas del error de estimación de los descriptores de movimiento, o analizar la robustez de los métodos con respecto al ruido. En estas imágenes, pueden emplearse como medidas, la desviación entre los desplazamientos estimado y real [NETR79] o medidas angulares del error de estimación [BARR94].

Para imágenes reales, la cantidad más usada es la entropía del campo de desplazamiento, ya que proporciona una medida de la *suavidad* del mismo: una entropía baja representa un campo suave, con variaciones pequeñas de una región a otra de la imagen. La entropía se usa tanto para evaluar la calidad del campo como la posible compresión de datos. Para esta última, es deseable que se minimice tanto la DFD como la entropía del campo. Por otra parte, para una segmentación basada en movimiento, conviene tener un campo poco ruidoso, que equivale a una entropía baja.

Sin embargo, la entropía no es una medida concluyente acerca de la calidad: en un campo demasiado suave, se pierden las fronteras de movimiento, lo que complica la detección de las fronteras de las regiones en una segmentación. Por ello, la apreciación visual del campo sigue siendo una medida subjetiva necesaria y relevante.

**2. Calidad de la reconstrucción de la secuencia a partir del campo de desplazamiento estimado.** Esta se juzga tanto visual como cuantitativamente. Las medidas numéricas empleadas son el error cuadrático medio (EQM) y la relación señal a ruido pico (PSNR) entre cada trama de la secuencia original y de la secuencia reconstruida [BAAZ91a]. Para los algoritmos pel-recursivos en particular, una medida comúnmente empleada es el porcentaje de pixels no compensados [WALK86], [BAAZ91a], [WANG92], que también puede usarse como información para evaluar la calidad de campo.

**3. Compresión.** La medida típica usada con algoritmos pel-recursivos es el valor RMS del error de predicción (RMSPE), que indica el tamaño de la información a transmitir [BIEM87]. Otra medida similar es la entropía del error de predicción [MOOR87], [WANG92], [BAAZ91a]. Cuando se usa un estimador de movimiento formando parte de un sistema de compresión completo, es común reportar el número total de bits transmitidos (datos y direcciones) para cada trama de la secuencia [MUSM85], [WALK87].

**4. Complejidad computacional.** Estas medidas pueden incluir desde el número de operaciones aritméticas por pixel, hasta la evaluación de las propiedades del algoritmo que hagan posible su implantación en diversas arquitecturas de cálculo [PAQU83]. En este trabajo se sigue el criterio normalmente empleado para los algoritmos pel-recursivos: el valor medio del número de iteraciones efectuadas por pixel [WALK86], [BAAZ91a].

En [GATI96] se incluyen las expresiones matemáticas para las medidas usadas.

## 4.2 Secuencias de prueba.

1. Secuencia *Camarógrafo*. Secuencia sintética (150x128, 8 bits/pixel). Se simula movimiento puramente translacional, sin ruido. El desplazamiento es idéntico para todos los pixels de la imagen, y puede variarse para obtener cualquier dirección y

magnitud. En la figura 4.1 se muestran dos tramas de esta secuencia, y su diferencia temporal.

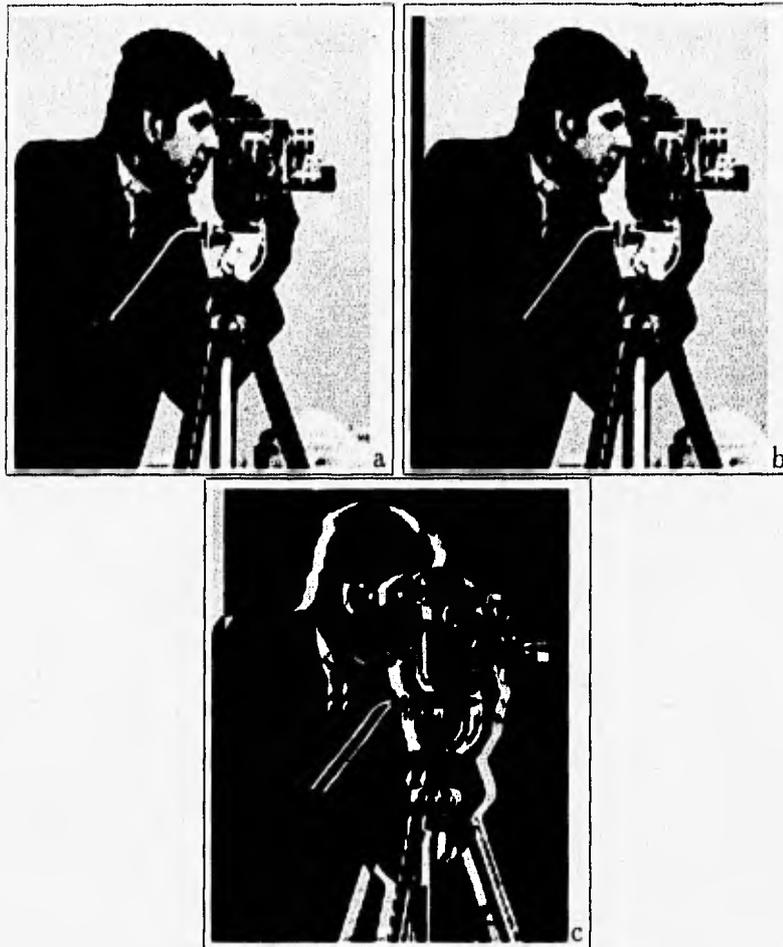


Figura 4.1: Secuencia *Camarógrafo*. a) Trama número 1. b) Trama número 2. c) Valor absoluto de la  $FD$  entre ambas tramas.

2. Secuencia *Seqmilt*. Secuencia sintética (256x256, 8 bits/pixel). Presenta cuatro objetos con texturas variadas, experimentando translación, rotación y divergencia puras, sobre un fondo texturado (tablero de ajedrez). Se presentan problemas de oclusión entre los objetos y ruido aditivo. El movimiento se sintetiza utilizando un modelo lineal simplificado de 4 parámetros  $\Theta_{MLS}$ .

3. Secuencia *Miss América*. Secuencia real típica de ambiente de videotelefonía, que muestra la cabeza y los hombros de una mujer en movimiento, sobre un fondo fijo (288 x 360, 8 bits/pixel). Es importante comentar el pobre contraste existente entre el fondo de la escena y el cabello de la mujer, además de efectos de

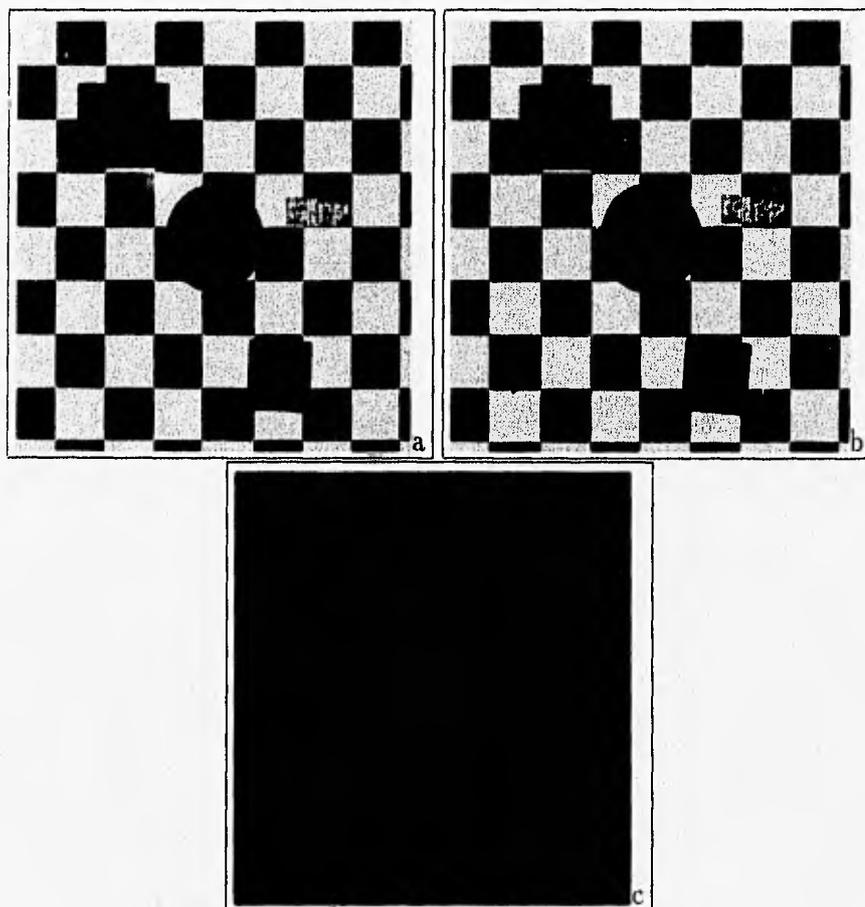


Figura 4.2: Secuencia *Seqmilt*. a) Trama número 1. b) Trama número 2. c) Valor absoluto de la  $FD$  entre ambas tramas.

sombras (figura 4.3).



Figura 4.3: Secuencia *Miss América*. a) Trama número 3. b) Trama número 4. c) Valor absoluto de la  $FD$  entre ambas tramas (amplificado por 10).

4. Secuencia *Interview*. Secuencia real de televisión (536x674, 8 bits/pixel). Presenta una escena de interiores donde una mujer se pone de pie, mientras la cámara se mueve hacia arriba, intentando seguir el desplazamiento de la mujer. (figura 4.4).

La figura 4.5 muestra el valor RMS de la diferencia interimagen ( $FD$ ) de las secuencias de prueba *Seqmill*, *Miss América* e *Interview*. De ella se puede decir:

- En *Seqmill*, el movimiento es más o menos constante durante toda la secuencia.
- En *Miss América* el movimiento es variable, pero siempre de pequeña magnitud. Es la secuencia con menos movimiento.
- *Interview* tiene más movimiento que las otras dos secuencias; el movimiento es mayor conforme la secuencia progresa.



Figura 4.4: Secuencia *Interview*. a) Trama número 6. b) Trama número 7. c) Valor absoluto de la  $FD$  entre ambas tramas (amplificado por 1.5).

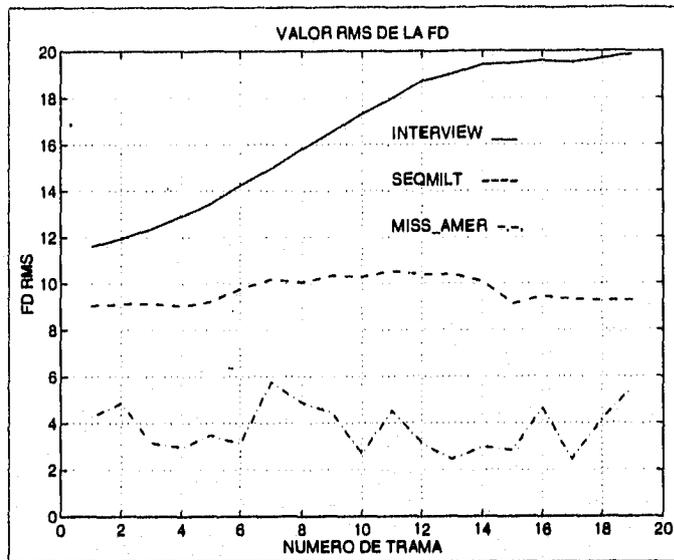


Figura 4.5: Valor RMS de la *FD* de las secuencias *Seqmilt*, *Miss América* e *Interview*.

### 4.3 Resultados.

En este trabajo se eligieron para el análisis los dos algoritmos pel-recursivos más representativos: Walker y Rao (WR) y Biemond et al (BD), junto con los dos métodos combinados WR-BD propuestos por Baaziz (BZ1, BZ2). Todos los métodos se aplicaron a una sola resolución.

La primera secuencia de prueba se presenta para poner en evidencia algunas de las diferencias básicas entre los métodos originales WR y BD. Para las tres secuencias restantes, se presenta en primer término la información referente a los campos de desplazamiento obtenidos con los métodos originales y combinados; posteriormente, se muestran los resultados correspondientes a la compensación de movimiento (calidad de reconstrucción de las secuencias).

#### 4.3.1 Movimiento translacional puro.

El caso más simple a simular es un movimiento translacional puro para toda la imagen (secuencia *Camarógrafo*). Este tipo de simulación es utilizada comúnmente [BIEM87], [BARR94]. La figura 4.6 muestra los campos obtenidos con los métodos *originales* (WR y BD) para un desplazamiento horizontal de 5 pixel/trama ( $d = (5,0)$ ). Como se aprecia, con el método WR no se alcanza la convergencia mas que en ciertas zonas de la imagen, mientras que con BD el desplazamiento converge después de un cierto número de pixels, y el campo es en general mucho más coherente. Si se aumenta el número de iteraciones por pixel, mejora la convergencia

en ambos métodos, pero el campo generado por BD siempre es más congruente. Los resultados son similares cuando se incrementa el valor del desplazamiento a 7 pixels/trama. Para desplazamientos mayores (hasta 8-9 pixels/trama), se necesita un mayor número de iteraciones. Más allá, el algoritmo BD muestra problemas de convergencia cada vez más severos. Esta es una limitación inherente a todos los métodos diferenciales de estimación de movimiento.

La tabla 4.1 muestra los valores esperados ( $\mu_{dx}, \mu_{dy}$ ), las desviaciones estándar ( $\sigma_{dx}, \sigma_{dy}$ ) y las entropías de ambos campos de desplazamiento ( $H_{dx}, H_{dy}, H_{total}$ ). Siguiendo a [BARR94], se incluye también una medida angular promedio del error entre el desplazamiento correcto y el estimado ( $\theta_{error}$ ). Es evidente que el campo obtenido por BD es mejor en cuanto a precisión y convergencia en la estimación.

	Método BD	Método WR
$\mu_{dx}$ (pixels)	4.80	1.80
$\sigma_{dx}$ (pixels)	1.22	2.34
$\mu_{dy}$ (pixels)	0.26	0.07
$\sigma_{dy}$ (pixels)	1.40	1.44
$H_{dx}$ (bits)	0.85	5.77
$H_{dy}$ (bits)	0.69	5.21
$H_{total}$ (bits)	1.54	10.99
$\theta_{error}$ (grados)	4.90	52.16

Tabla 4.1. Valores esperados, desviaciones estándar y entropías de los campos generados con los metodos WR y BD:  $\mathbf{d} = (5, 0)$ . Secuencia *Camarógrafo*.

Un segundo caso se ilustra en la figura 4.7, para un desplazamiento  $\mathbf{d} = (3, 3)$ . Nuevamente se observa que el campo generado con BD es más coherente y más precisamente estimado que el obtenido con WR. En la tabla 4.2 se muestran los resultados obtenidos.

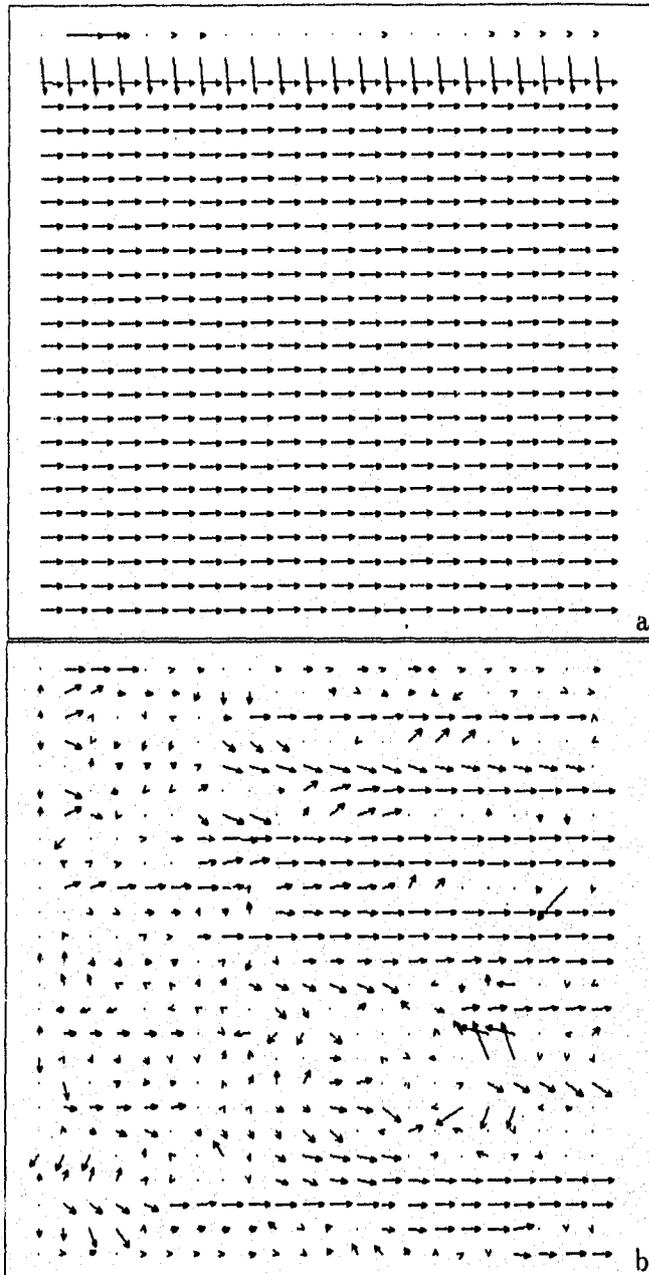


Figura 4.6: (a) Campo translacional ( $\mathbf{d} = (5,0)$ ) obtenido con el método BD. (b) Campo obtenido con el método WR. Los campos se muestran submuestreados cada 6 pixels. Para los dos métodos se hicieron como máximo 5 iteraciones por pixel, y se usó como predicción el desplazamiento calculado para el pixel anterior. Para BD,  $N=7$ ,  $\mu=100$ .

	Método BD	Método WR
$\mu_{dx}$ (pixels)	3.03	1.03
$\sigma_{dx}$ (pixels)	0.94	1.71
$\mu_{dy}$ (pixels)	2.90	0.67
$\sigma_{dy}$ (pixels)	0.53	1.65
$H_{dx}$ (bits)	0.95	5.52
$H_{dy}$ (bits)	0.79	5.58
$H_{total}$ (bits)	1.74	11.11
$\theta_{error}$ (grados)	2.47	57.2

Tabla 4.2. Valores esperados, desviaciones estándar y entropías de los campos generados con los metodos WR y BD:  $d = (3, 3)$ . Secuencia *Camarógrafo*.

Aunque esta situación es demasiado simple, pueden comenzarse a ver diferencias entre los dos algoritmos pel-recursivos básicos, en cuanto a precisión numérica, y coherencia del campo.

### 4.3.2 Movimiento sintético variado.

Una secuencia sintética con distintos movimientos, no puramente translacional, puede arrojar más información sobre el comportamiento de los algoritmos. Para esto se utiliza la secuencia *Seqmilt*.<sup>1</sup> Los objetos en la escena son:

Objeto	tx	ty	k	$\theta$	Movimiento puro
Fondo	0.0	0.0	0.0	0.0	Sin movimiento
Rectángulo 1	4.0	1.0	0.0	0.0	Traslación
Rectángulo 2	-2.0	-1.0	0.0	0.0	Traslación
Círculo	0.0	0.0	0.025	0.0	Divergencia
Cuadrado	0.0	0.0	0.0	0.04	Rotación

Tabla 4.3. Parámetros de la secuencia *Seqmilt*.

#### 4.3.2.1. Campos de desplazamiento estimados.

##### Métodos originales

La figura 4.8(a) muestra el campo obtenido para el método BD (tramas 1 y 2). Se puede apreciar lo siguiente:

<sup>1</sup>Los parámetros usados en todas las pruebas reportadas en este capítulo fueron  $N=7$ ,  $\mu=100$ ,  $N_{vec}=5$ ,  $T_{FD}=T_{DFD}=2$ ,  $it_{max}=1$  (según fueron aplicables o no a cada método), a menos que se indique otra cosa.

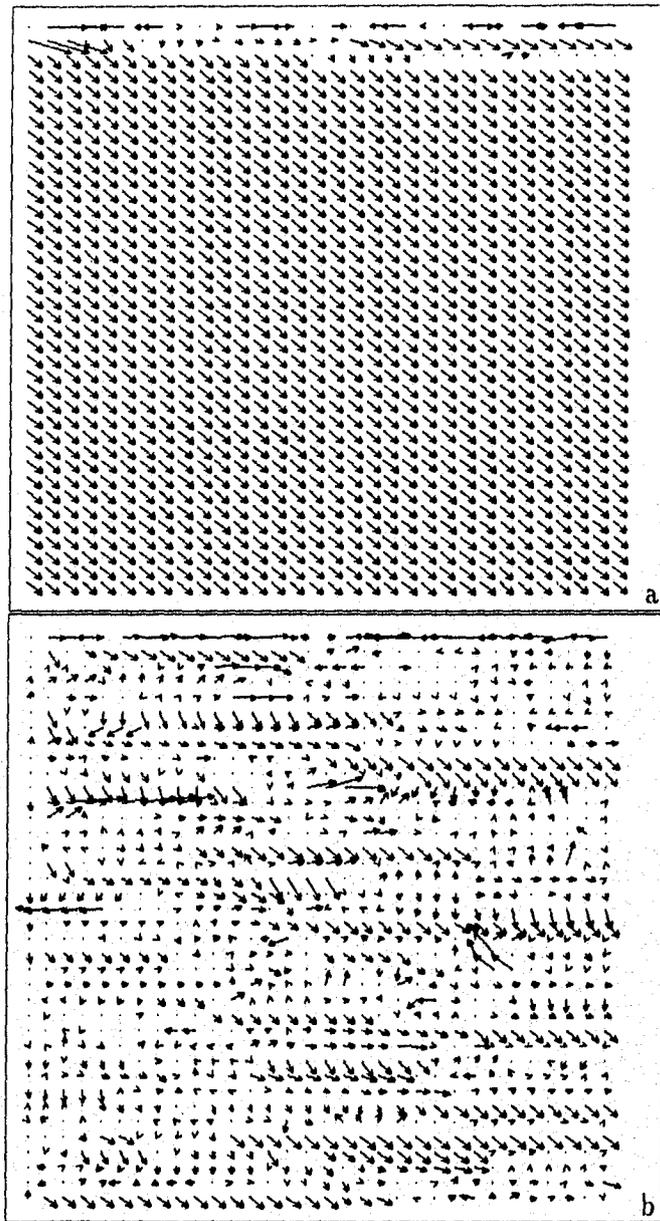


Figura 4.7: (a) Campo translacional con movimiento vertical y horizontal ( $\mathbf{d} = (3,3)$ ) obtenido con el método BD. (b) Campo obtenido con el método WR. Las condiciones de prueba son las mismas que para la figura 4.6. Los vectores se muestran submuestreados cada 4 pixels.

- El movimiento cualitativamente es bien estimado en general. Pueden apreciarse los movimientos de translación, rotación y divergencia de cada objeto. Cuantitativamente también los resultados son buenos. Sin embargo, la estimación precisa de movimientos distintos al translacional es más difícil de alcanzar (recuérdese que BD parte de suponer sólo translación pura).
- Las regiones de descubrimiento no son bien estimadas (esto es particularmente notable en el rectángulo grande).
- Se dijo en el capítulo anterior que en los métodos pel-recursivos existe un par de restricciones de suavizado local implícitas: la elección del desplazamiento de algún vecino como predicción, y la lógica de funcionamiento de los algoritmos. Sin embargo, en el método BD, la restricción de suavizado más importante proviene de efectuar la estimación considerando información en toda una vecindad espacial (las  $N$  observaciones en la sección 3.4). Estos factores son responsables del efecto de *inercia en el movimiento* en la dirección de barrido (los vectores de movimiento se propagan con un mismo valor durante varios pixels aun y cuando el movimiento real sea diferente) que se aprecia en algunas regiones (generalmente en fronteras de movimiento). Este efecto es notorio en todos los objetos.
- El algoritmo tolera bien el ruido. Existen pequeñas regiones en el fondo en donde se estima un movimiento erróneo debido al ruido, pero esto no afecta la estimación del movimiento de los objetos. Sin embargo, el hecho de considerar la vecindad  $N$  para la estimación, que mejora el resultado cuando todos los pixels en la vecindad pertenecen a un mismo objeto en movimiento, ocasiona errores en la estimación en las fronteras de movimiento.
- En general, una sola iteración por pixel es suficiente para la obtención de campos vectoriales adecuados.

En la figura 4.8(b) se muestra el campo obtenido para el método WR. En este caso se observa que:

- El campo permite distinguir el movimiento de los dos rectángulos, pero no muestra coherencia alguna, ya que el ruido domina la estimación. Los movimientos de divergencia y rotación difícilmente se pueden apreciar. En una palabra, este campo no corresponde con la realidad.
- ¿Qué es lo que está ocurriendo? Hay que recordar el concepto original del algoritmo WR. Este es esencialmente un método de optimización *local*, donde se minimiza una determinada función de costo ( $DFD$ ) empleando técnicas de gradiente. Ahora bien, la  $DFD$  es ante todo un criterio de *reconstrucción* de secuencias por compensación de movimiento, pero *no* es suficiente para asegurar que el campo obtenido corresponda a algún movimiento físico real

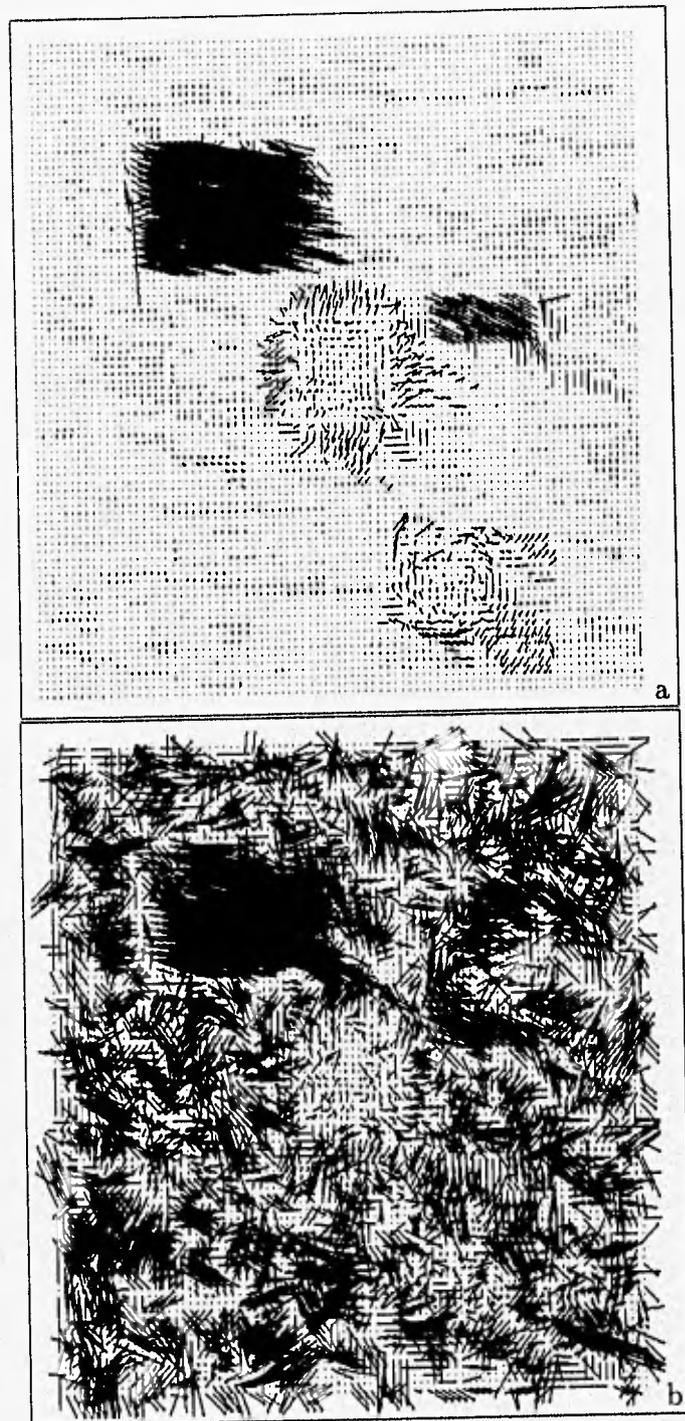


Figura 4.8: Campos de desplazamiento estimados para la secuencia *Seqmill*. Tramas 1 y 2. (a) Método BD. (b) Método WR. La escala de los vectores de movimiento se ha amplificado por 4. El campo está submuestreado cada 3 pixels.

[NICO92a]. Mientras que en el método BD se impone, como característica de suavizado en la estimación del campo, la hipótesis de que las  $N$  observaciones (puntos en la vecindad espacial) pertenecen a un mismo objeto de la escena en movimiento, esto no ocurre en WR, donde la estimación es puntual. Así, cuando directamente la  $DFD^2$  es la función de energía a minimizar, sin restricciones de suavizado adicionales, muchos valores distintos de desplazamiento pueden satisfacer el criterio de minimización, especialmente en ausencia de información de vecindad local, como en una zona de intensidad homogénea. Este es justamente el efecto que se observa en la figura 4.8(b): existe ruido en toda la imagen, pero el fondo es más o menos homogéneo dentro de cada cuadro (claro u oscuro) del tablero de ajedrez. Cuando se aprecia el campo de desplazamiento con los vectores en su escala original, se observa claramente que el desplazamiento estimado para cada pixel del fondo de la imagen tiende a quedar confinado dentro del cuadro del tablero al que pertenece cada pixel. Se puede concluir para esta secuencia que la estimación del movimiento por WR es altamente susceptible a corromperse por ruido [WANG92].

- Para terminar, por lo mencionado anteriormente, si bien el método WR no generó un campo que correspondiera al movimiento físico real, sí debe permitir una buena calidad de reconstrucción de la secuencia.

La figura 4.9 muestra el comportamiento de la entropía del campo estimado para los métodos pel-recursivos para las primeras 20 tramas de la secuencia *Seqmill*. Como se esperaba de las figuras anteriores, la  $H_{total}$  obtenida con BD es mucho más baja que la obtenida con WR. Además, las entropías para cada método se mantienen aproximadamente constantes, lo cual indica que los campos de movimiento conservan más o menos las mismas características a lo largo de la secuencia.

Un parámetro típico considerado en estos métodos es el de pixels no compensados, que representa el porcentaje de puntos para los cuales no se satisface el criterio de convergencia ( $|DFD| < T_{DFD}$ ). Este porcentaje es un indicador de cuándo la información de movimiento es poco precisa, en el sentido de *reconstrucción de la imagen*. Depende evidentemente del tamaño del umbral impuesto al algoritmo. La figura 4.10 muestra las imágenes de pixels no compensados, y la tabla 4.4 los resultados para los métodos originales y los combinados.

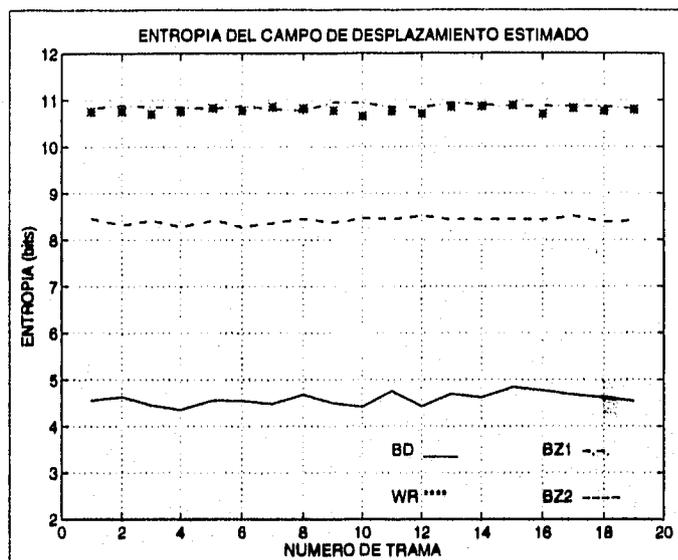


Figura 4.9: Entropía de los campos de desplazamiento estimados por los métodos pel-recursivos. Secuencia *Seqmill*.

	<b>BD</b>	<b>WR</b>	<b>BZ1</b>	<b>BZ2</b>
% pixels compensados (recursividad)	80.11	85.36	85.63	84.15
% pixels compensados (detección de mov.)	0.69	2.22	2.23	1.92
% pixels compensados (iteratividad)	1.82	5.44	5.64	6.43
% pixels no compensados	17.36	6.96	6.48	7.49
promedio de iteraciones/pixel	0.19	0.12	0.24	0.25

Tabla 4.4. Porcentajes de compensación obtenidos para los métodos originales (BD, WR) y los combinados (BZ1, BZ2). Secuencia *Seqmill*.

De la figura 4.10 se puede ver que los pixels para los cuales los métodos no convergen ocurren en regiones ruidosas y en las fronteras de movimiento de los objetos. También es interesante observar que el método WR presenta mejores características de compensación, debido a la libertad que tiene (estimación puntual) para encontrar un mínimo local para la *DFD*. El método BD, que genera campos mas suaves (y congruentes con el movimiento aparente), presenta por tanto mayores restricciones para satisfacer el criterio de convergencia.

En secuencias reales, también se encuentra que los pixels no compensados se concentran en fronteras de movimiento y regiones ruidosas de la imagen. Por tanto, aunque los pixels no compensados no sean la mejor medida de confianza en cuanto a congruencia del campo con la realidad física, sí representan información útil en regiones donde regularmente la estimación de movimiento no es buena.

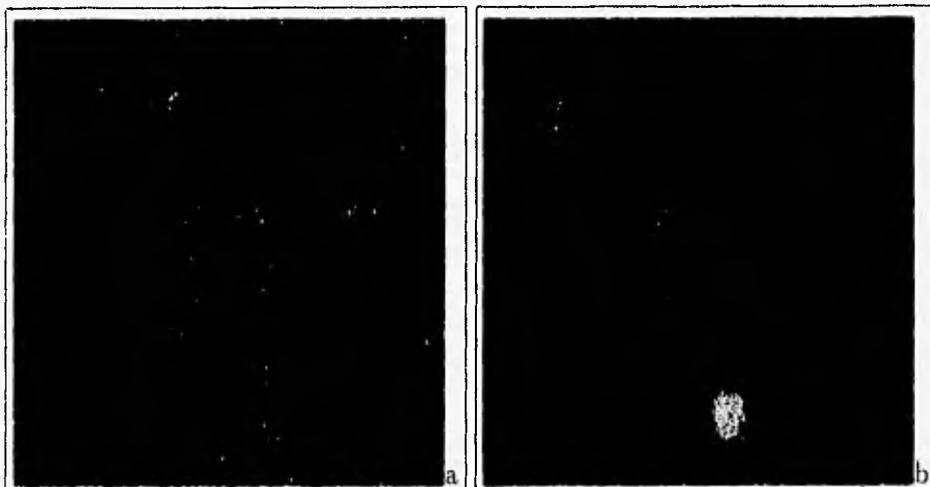


Figura 4.10: Imágenes de pixels no compensados para (a) BD y (b) WR. Secuencia *Seqmilt*.

#### Métodos combinados.

Una característica general de los métodos pel-recursivos es su dependencia del valor de predicción (vector de desplazamiento inicial)  $\mathbf{d}^0$ . Esto se debe a que la función de costo puede contener muchos mínimos locales, y en un método de optimización local, distintas condiciones iniciales pueden originar distintas soluciones [NICO92b]. En los métodos pel-recursivos, el  $\hat{\mathbf{d}}^0 = \hat{\mathbf{d}}^{l-1}$  se elige como aquel que minimice localmente la *DFD* [BAAZ91b]. Esto se discute con detalle en una subsección posterior, pero es pertinente mencionarlo aquí porque determina en buena forma el comportamiento de los métodos combinados BD-WR.

En la figura 4.9 se muestra, además del comportamiento de la entropía para los algoritmos originales, el obtenido para las combinaciones propuestas por Baaziz. No hay que perder de vista que estos métodos combinados buscan mejorar la calidad de reconstrucción de la secuencia.

Como se aprecia, el método BZ1 tiene una entropía similar a la de WR (las características de ambos campos son similares), y su porcentaje de pixels no compensados es residualmente menor. La explicación se encuentra observando que el método BZ1 se basa en la minimización de la *DFD* usando el método (WR o BD) que resulte mejor en cuanto a reconstrucción en cada pixel, con lo que habrá vectores generados por ambos métodos en cualquier región de la imagen. Por otra parte, ya se vio que un campo WR es más ruidoso (en el sentido de coherencia del campo) que un campo BD.

Ahora bien, como se ve de la tabla 4.4, el método WR original converge para

un número mayor de pixels que el BD original, lo que hace pensar que, en el método BZ1, a un gran porcentaje de la imagen se le debe aplicar el método WR. En la práctica se encontró que entre un 65% y un 75% de los pixels son compensados por este último. El resultado final es que el campo BZ1 tiende a *contaminarse* por ruido en la estimación porque:

1. Cada  $\hat{d}^i$  estimado por WR introduce algún ruido.
2. Estos  $\hat{d}^i$  pueden llegar a elegirse como la predicción  $\hat{d}^0$  para algún pixel vecino, y generar también ruido.
3. Por la lógica del algoritmo (pruebas de recursividad y detección de movimiento), estos vectores se pueden propagar y convertirse en la estimación final para muchos otros pixels.
4. A los pixels no compensados, se les puede asociar como estimación de desplazamiento final un vector ruidoso (sección 4.3.5.8).

Por otra parte, la entropía para el método BZ2 muestra un valor intermedio entre los de BD y WR. Asimismo, el porcentaje de pixels no compensados es menor que el obtenido por BD, pero mayor que el correspondiente a WR. El campo de desplazamiento se aprecia ruidoso, por las razones antes expuestas, pero menos que en el método BZ1, porque en BZ2 siempre se realiza primero la estimación con BD, y en los casos en que no hay convergencia, se aplica WR. La figura 4.11 muestra los campos para los metodos combinados.

#### **4.3.2.2. Calidad de la reconstrucción de la secuencia.**

La figura 4.12 muestra la PSNR de las primeras 20 tramas compensadas por movimiento de la secuencia *Seqmilt*, utilizando los cuatro métodos analizados. Como se observa, se producen en todos los casos secuencias con muy buena calidad (entre 38.2 y 42.7 dB), y el juicio visual también es favorable. Estos resultados concuerdan con el hecho de que la filosofía que rige a esta familia de métodos es justamente la minimización local del error de reconstrucción.

En la figura 4.13 se muestran las imágenes de error de reconstrucción obtenidas para cada método correspondientes a las dos primeras tramas de la secuencia. Se puede ver que el error de reconstrucción más alto se presenta en las fronteras de movimiento, en regiones con ruido, y en particular en el círculo en las zonas donde el desplazamiento debido a la divergencia es mayor. El error en BD es mayor que en los otros métodos. Por último, en WR, BZ1 Y BZ2 se nota muy ligeramente el error en la cuadrícula del tablero de ajedrez.

En resumen, todos los métodos son satisfactorios en cuanto a calidad de reconstrucción, siendo BD el que presenta resultados menos buenos. Es importante

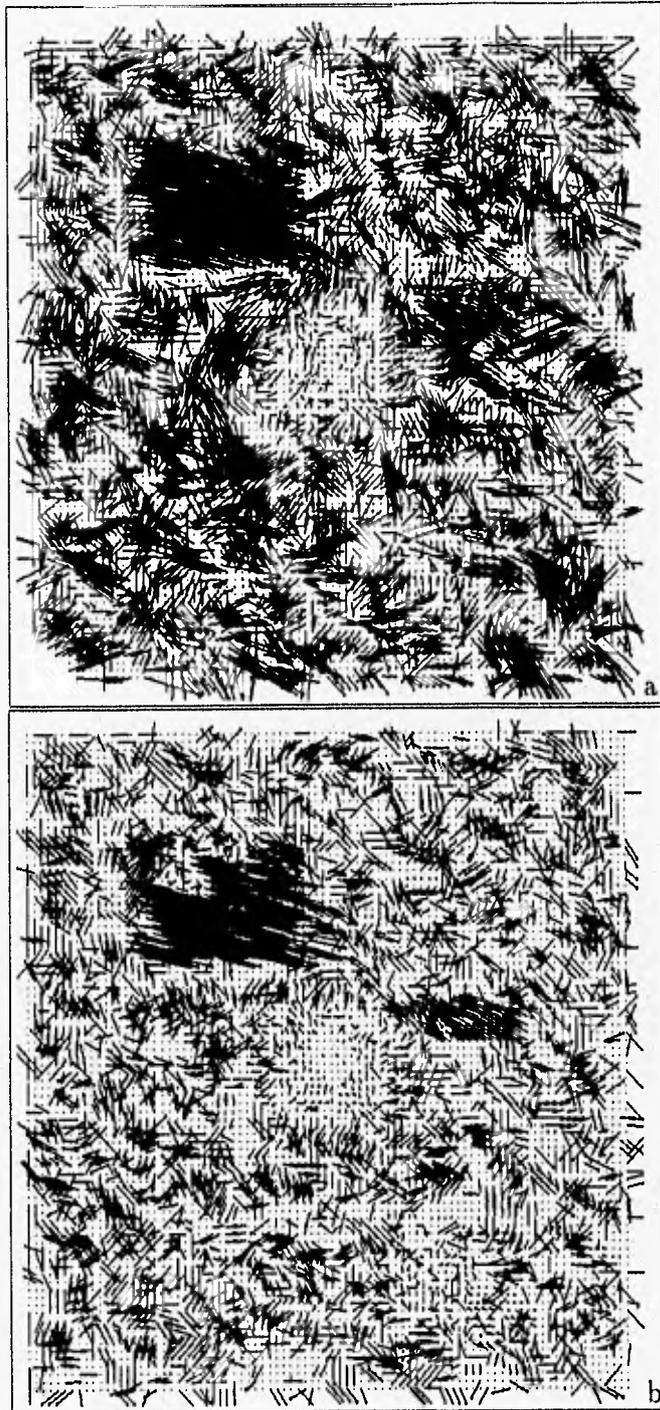


Figura 4.11: Campos de desplazamiento estimados para la secuencia *Seqmill*. Tramas 1 y 2. Métodos combinados BD-WR. (a) BZ1. (b) BZ2. La escala de los vectores de movimiento se ha amplificado por 4. El campo está submuestreado cada 3 pixels.

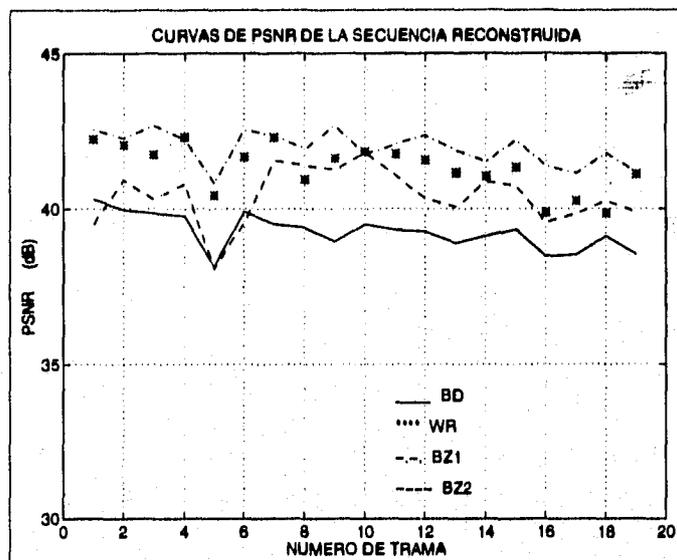


Figura 4.12: PSNR de la secuencia *Seqmilt* reconstruida.

hacer notar que se ha mostrado que campos de desplazamiento bastante diferentes pueden producir resultados visuales semejantes.

En este punto, debe quedar claro que existe un compromiso entre la calidad de la reconstrucción y la coherencia del campo de desplazamiento. Por un lado, los campos generados a partir de restricciones fuertes de suavizado como los desarrollados en [HORN86], o las variantes que incluyen hipótesis de continuidad local [NAGE86], permiten obtener campos visualmente coherentes, que los hacen atractivos para análisis cualitativo de movimiento, pero no para codificación. Esto se debe a que las restricciones de suavizado implican casi siempre un incremento en el error de reconstrucción (el suavizado en las fronteras de movimiento es excesivo) [NICO92b]. Por otra parte, ya se vio que el método WR genera una muy buena calidad de reconstrucción de la imagen, pero sus características de continuidad no son buenas. El método BD parece presentar una combinación de ambas cualidades. Es claro que este comportamiento debe ser valorado en secuencias reales. Tales resultados se presentan en las siguientes secciones.

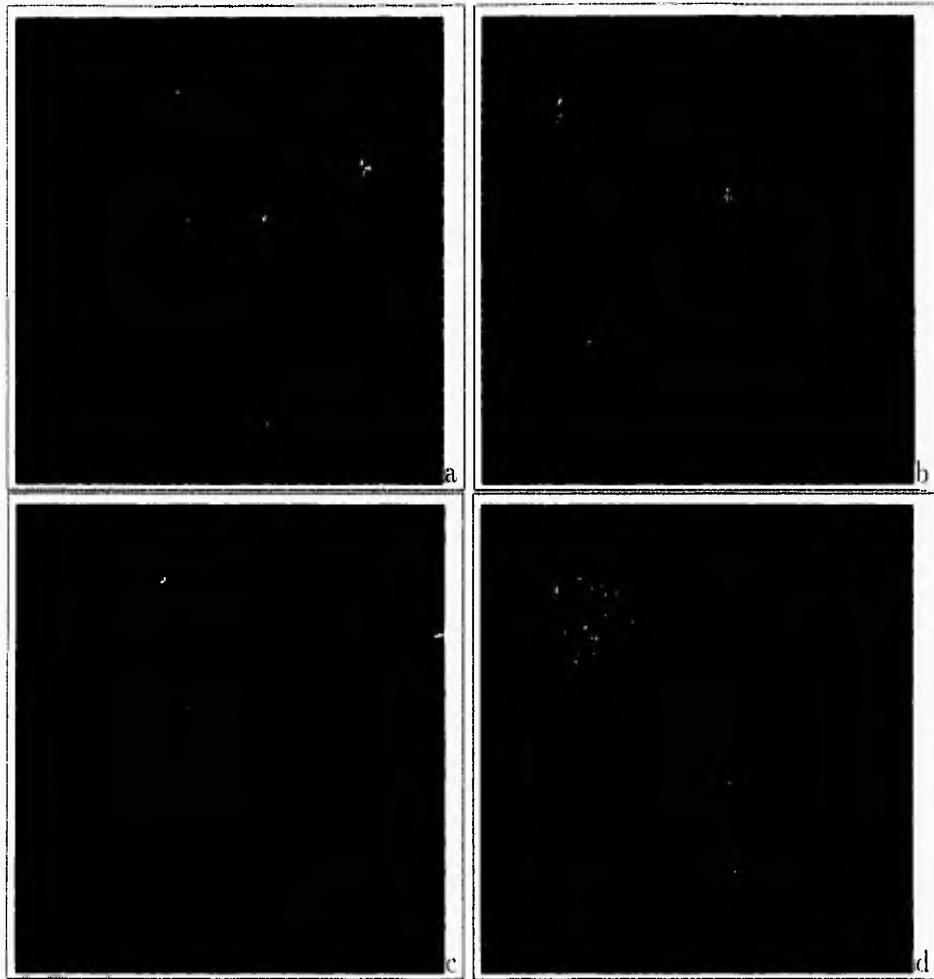


Figura 4.13: Imágenes de error de reconstrucción. Secuencia *Seqmill*. Tramas 1 y 2. (a) BD, (b) WR, (c) BZ1, (d) BZ2. El error aparece amplificado por 25 para su visualización.

### 4.3.3 Secuencia real Miss América.

#### 4.3.3.1. Campos de desplazamiento estimados.

La secuencia *Miss América* es interesante porque contiene movimiento distinto a lo largo de la misma: en ocasiones el movimiento de la mujer es global; en otras, el movimiento más importante se observa en los ojos y la boca, mientras que la cabeza y torso permanecen casi inmóviles. Este último caso se presenta en las tramas 3 y 4. La figura 4.14 muestra los campos estimados por BD y WR, y la 4.15 los campos estimados por BZ1 y BZ2 para este par de imágenes.

Se distingue en el campo producido por BD la presencia de mayor movimiento en dos regiones distintas: el rostro y el cuello de la mujer. El primer movimiento es real, y se concentra en los ojos, la boca y la oreja derecha, siendo de menor magnitud en el resto del rostro; el segundo movimiento importante es un efecto de sombra que origina un movimiento aparente. El movimiento en el resto de la escena es de mucha menor magnitud, aunque se alcanza a perfilar la silueta de la mujer. Por tanto, puede decirse que este método está realizando una estimación que concuerda con el movimiento aparente observado. De igual forma que en la secuencia *Seqmilt*, aparece el efecto de inercia de movimiento, y pequeñas regiones ruidosas, que no afectan la apariencia general del campo.

Por su parte, WR genera un campo que detecta también estas regiones, pero muestra poca continuidad. Se puede observar el ruido introducido por la estimación en la región del fondo (inmóvil), pero sobre todo en el torso y hombros de la mujer. Aquí también se repite el comportamiento observado en *Seqmilt*, aunque el efecto de corrupción del fondo de la escena por ruido en la estimación es menos importante.

En cuanto a los métodos combinados, el campo correspondiente a BZ1 se parece mucho al de WR, mientras que el de BZ2 aparece un poco menos ruidoso.

El otro tipo de movimiento importante en la secuencia *Miss América* es el movimiento global de la mujer (figura 4.16). La figura 4.17. muestra los campos estimados por WR y BZ para las tramas 38 y 39. Los resultados son elocuentes, pero de ellos se desprende un comentario interesante. Un observador sabe, en esta parte de la secuencia, que cada región en movimiento (cabeza, hombros) debe tener un movimiento aproximadamente similar para todos los pixels que la conforman. Sin embargo, las estimaciones no producen estos resultados en las regiones correspondientes a la porción superior del cabello de la mujer, o a partes de su torso. La razón de este hecho es que los métodos estiman el movimiento *localmente*, y como en las regiones mencionadas no ocurren cambios espacio-temporales en la función de intensidad, el desplazamiento aparente estimado es muy pequeño o nulo. Este efecto es más claro en BD que en WR, y es inherente a todos los métodos diferenciales de estimación de movimiento.

La entropía de los campos estimados para los métodos pel-recursivos para las

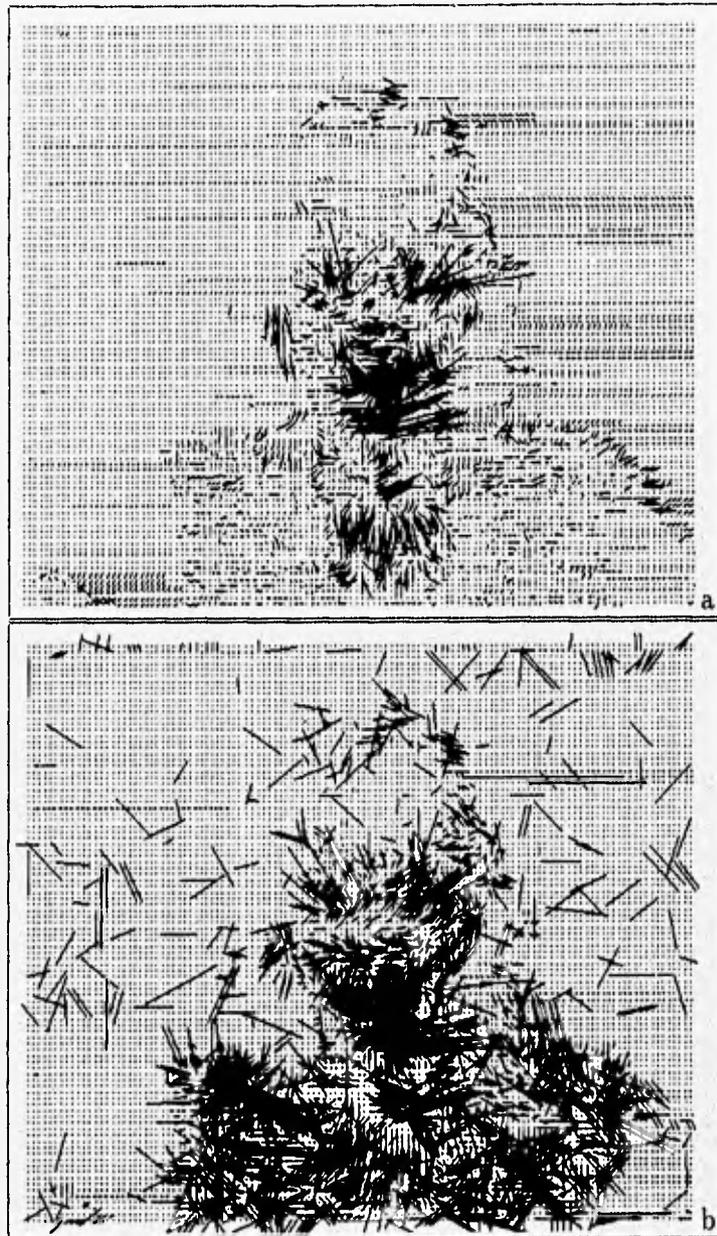


Figura 4.14: Campos de desplazamiento estimados para la secuencia *Miss América*. Tramas 3 y 4. (a) Método BD. (b) Método WR. La escala de los vectores de movimiento se ha amplificado por 10.

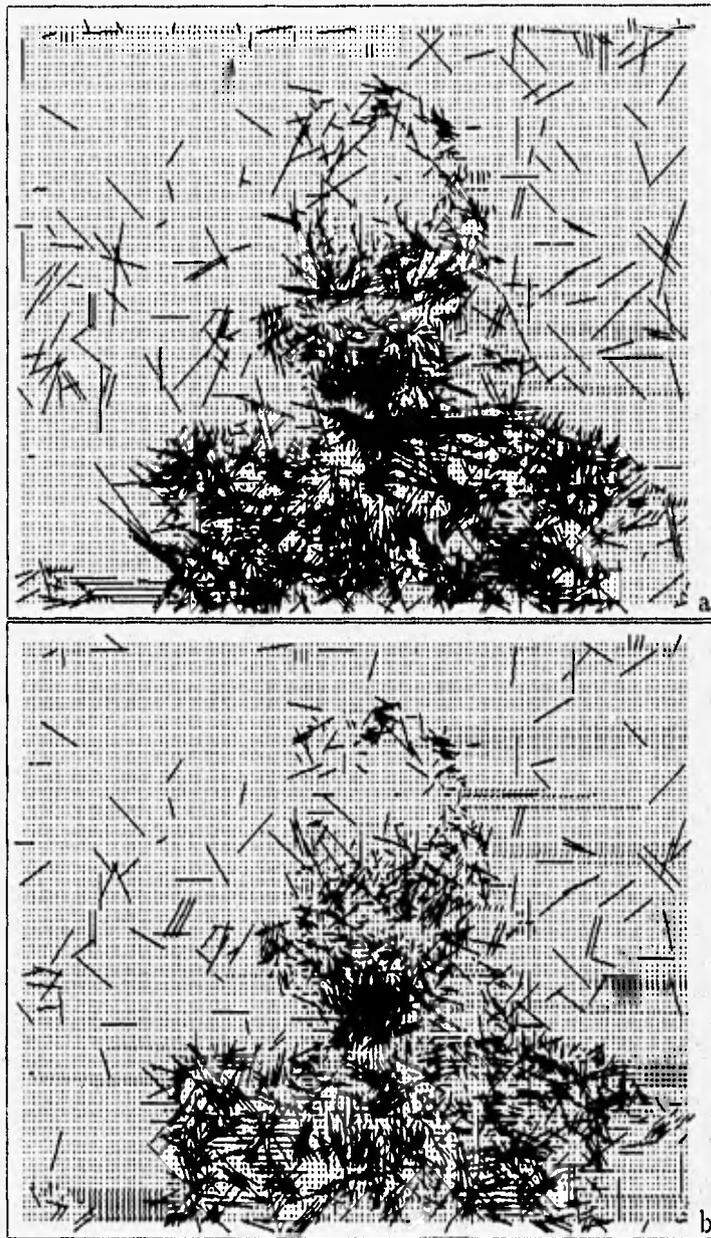


Figura 4.15: Campos de desplazamiento estimados para la secuencia *Miss América*. Tramas 3 y 4. (a) Método BZ1. (b) Método BZ2. La escala de los vectores de movimiento se ha amplificado por 10.

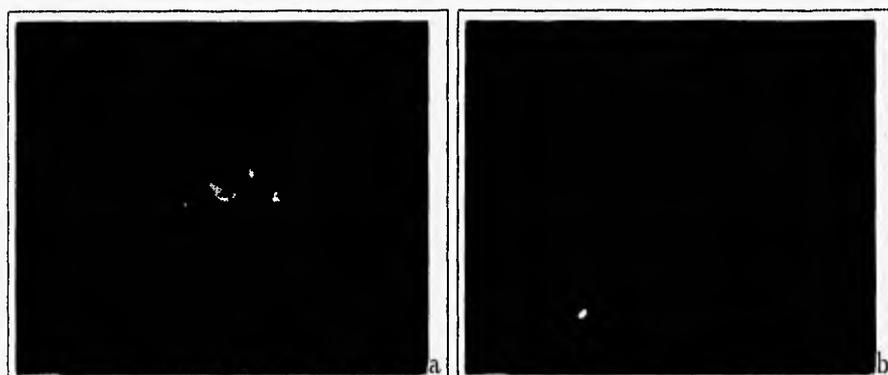


Figura 4.16: Secuencia *Miss América* a) Trama número 38. b) Trama número 39.

primeras 20 tramas de esta secuencia se muestra en la figura 4.18. En primer término, hay que hacer notar que la entropía de esta secuencia es inferior a la calculada para la secuencia *Seqmilt*. Por otra parte, la  $H_{total}$  obtenida con BD es nuevamente más baja que la obtenida con WR (alrededor de 1 bit durante toda la secuencia), pero en menor proporción que para el caso *Seqmilt*. Con respecto a los métodos combinados, se repite el comportamiento ya descrito con la secuencia sintética: BZ1 tiene una entropía prácticamente igual a la de WR, y BZ2 tiene un valor intermedio. Se muestra, por lo tanto, que el campo producido por BD es el que presenta mejores características en cuanto a continuidad.

En general, para esta secuencia con poco movimiento, los porcentajes de pixels no compensados son bajos. En la tabla 4.5 se muestran los resultados obtenidos para las tramas 3 y 4, y dan una idea de su comportamiento general a lo largo de la secuencia.

	<b>BD</b>	<b>WR</b>	<b>BZ1</b>	<b>BZ2</b>
% pixels compensados (recursividad)	91.50	92.86	92.97	92.90
% pixels compensados (detección de mov.)	0.27	0.63	0.65	0.45
% pixels compensados (iteratividad)	1.19	3.99	4.03	3.79
% pixels no compensados	7.03	2.50	2.33	2.85
promedio de iteraciones/pixel	0.08	0.06	0.13	0.12

Tabla 4.5. Porcentajes de compensación obtenidos para los métodos originales (BD, WR) y los combinados (BZ1, BZ2). Secuencia *Miss América*.

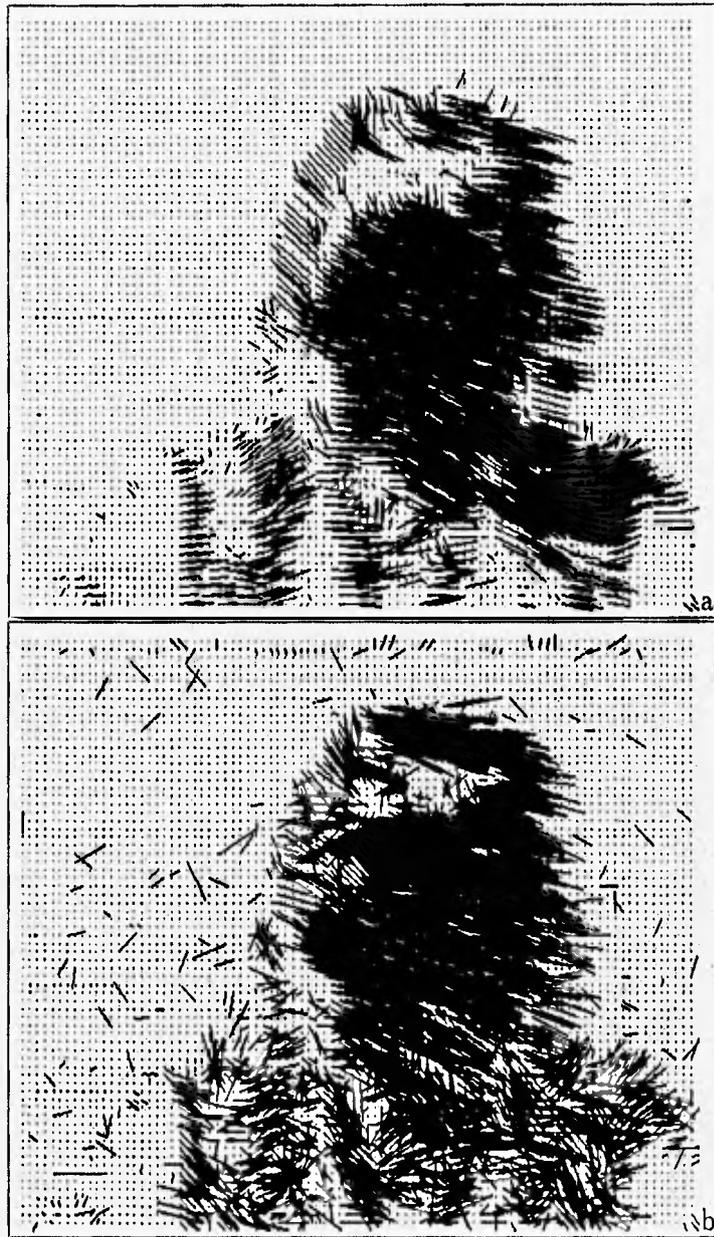


Figura 4.17: Campos de desplazamiento estimados para la secuencia *Miss América*. Tramas 38 y 39. (a) Método BD. (b) Método WR. La escala de los vectores de movimiento se ha amplificado por 5.

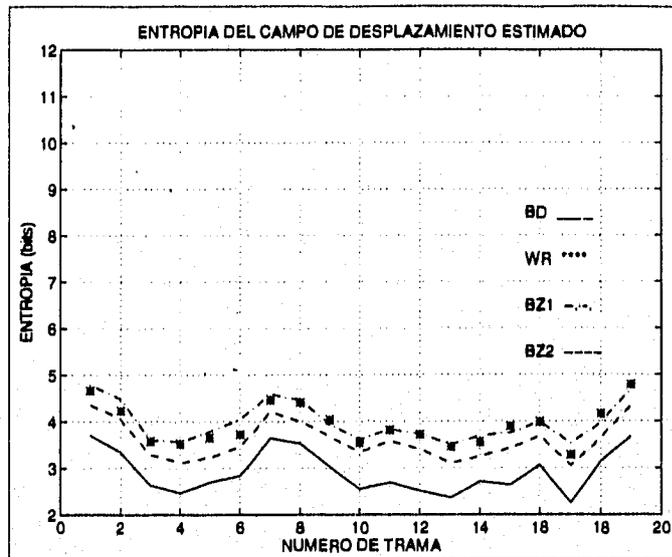


Figura 4.18: Entropía de los campos de desplazamiento estimados por los métodos pel-recursivos. Secuencia *Miss América*.

#### 4.3.3.2. Calidad de la reconstrucción de la secuencia.

La figura 4.19 muestra la PSNR de las primeras 20 tramas reconstruidas de la secuencia *Miss América*, utilizando los cuatro métodos. Se producen en todos los casos secuencias con excelente calidad numérica (entre 42.8 y 48.1 dB de PSNR), y visual. Esta gráfica muestra que los métodos pel-recursivos funcionan satisfactoriamente en secuencias con movimientos lentos.

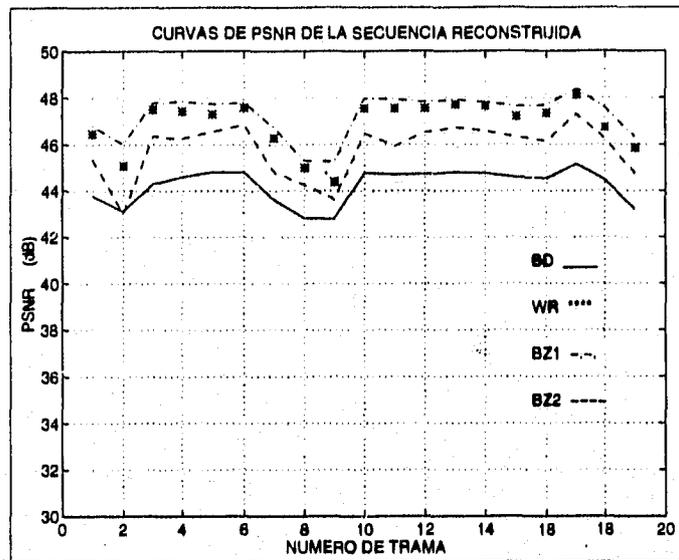


Figura 4.19: PSNR de la secuencia *Miss América* reconstruida.

#### 4.3.4 Secuencia real Interview.

La última secuencia de prueba utilizada tiene como principal característica la presencia de movimiento global de la escena, y la existencia de un movimiento distinto y de mayor magnitud (la mujer que se levanta).

##### 4.3.4.1. Campos de desplazamiento estimados.

Las figuras 4.20 y 4.21 muestran los campos estimados por los métodos pel-recursivos para las tramas de 6 y 7 de *Interview*. Puede apreciarse que, como en las secuencias anteriores, el método BD es el que presenta un campo más coherente. Puede distinguirse claramente el movimiento de la mujer, aunque existen errores en la estimación sobre todo en las fronteras de movimiento (fondo de la imagen y mujer). Este error es particularmente notable en la porción derecha de la cabeza de la mujer. El campo producido por WR nuevamente aparece más ruidoso, sobre todo en regiones correspondientes a las piernas de la mujer que no se mueve. Por su parte, el campo generado por BZ1 se asemeja al de WR, y el estimado por BZ2 es menos ruidoso que el de WR, pero con menos coherencia que el de BD. Estos resultados coinciden con lo encontrado con las dos secuencias anteriores.

La entropía de los campos estimados para las primeras 20 tramas de *Interview* se muestra en la figura 4.22. En primer lugar, se aprecia que la entropía estimada para esta secuencia es mayor que para las dos secuencias anteriores, y que además, muestra una tendencia creciente. Hay que recordar aquí el comportamiento de la

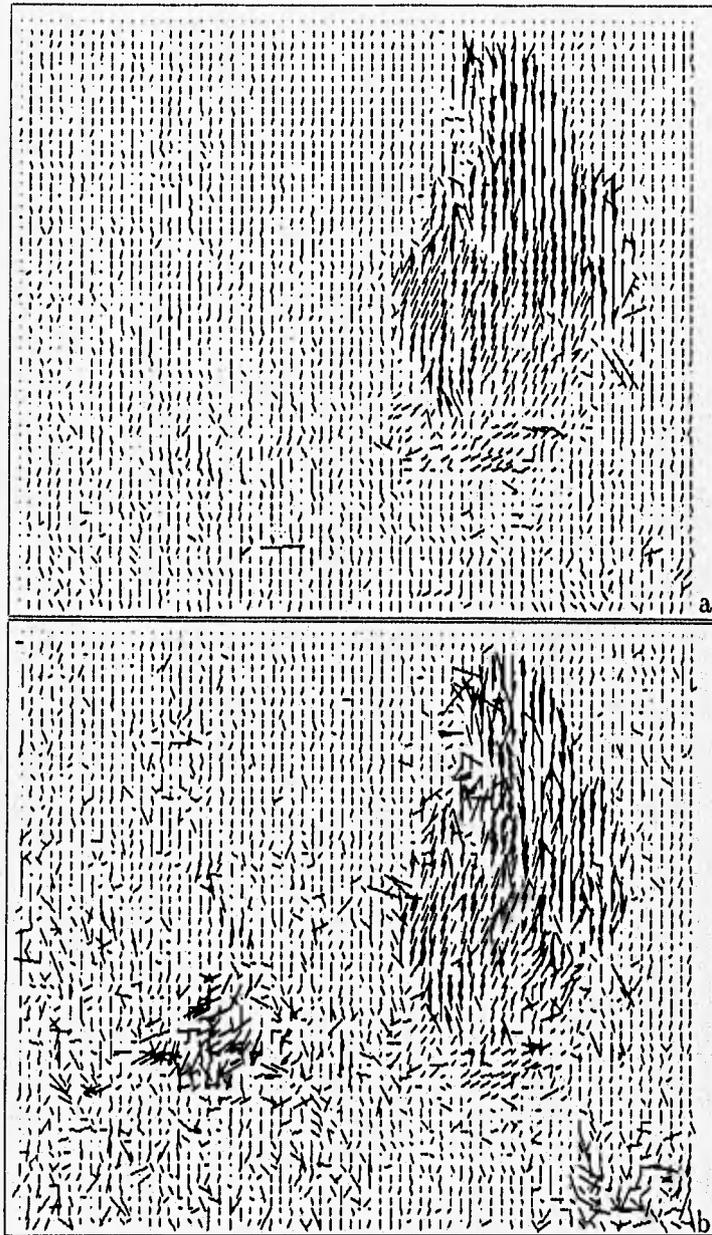


Figura 4.20: Campos de desplazamiento estimados para la secuencia *Interview*. Tramas 6 y 7. (a) Método BD. (b) Método WR. La escala de los vectores de movimiento se ha amplificado por 5. Para esta secuencia,  $N_{vec}=9$ .

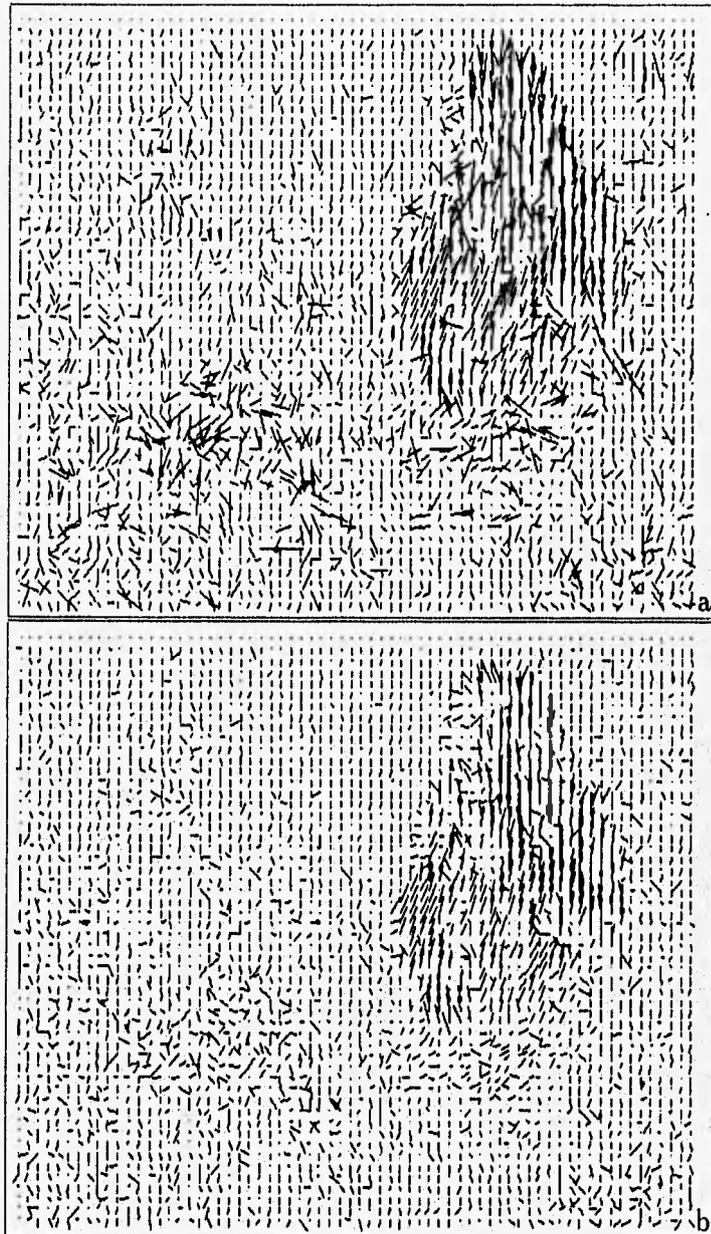


Figura 4.21: Campos de desplazamiento estimados para la secuencia *Interview*. Tramas 6 y 7. (a) Método BZ1. (b) Método BZ2. La escala de los vectores de movimiento se ha amplificado por 5.

FD de esta secuencia (figura 4.5), donde resulta evidente que el movimiento es creciente en magnitud. La entropía estimada aumenta conforme la secuencia avanza por dos causas:

1. El movimiento aparente de la mujer se hace más complejo (variado): a un campo más rico en información le corresponde una entropía mayor.
2. El movimiento global (cámara en movimiento) aumenta en magnitud, y se puede introducir algo de ruido en la estimación porque los desplazamientos mayores son más difíciles de estimar (esta característica es inherente a todos los métodos diferenciales).

En cuanto al comportamiento de cada método, la  $H_{total}$  obtenida con BD es nuevamente la más baja (un promedio de 0.6 bits menor con respecto a BZ2 y 1.6 bits menor con respecto a WR y BZ1 para toda la secuencia). El comportamiento comparativo de los métodos es similar al descrito con las dos secuencias de prueba anteriores.

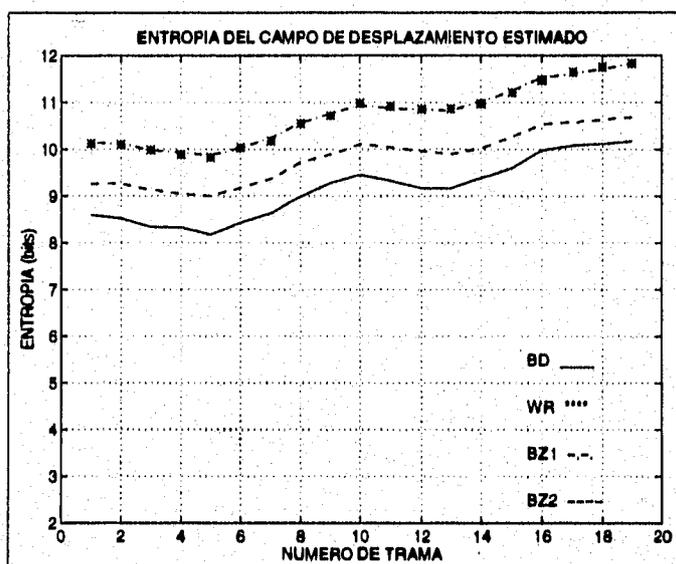


Figura 4.22: Entropía de los campos de desplazamiento estimados por los métodos pel-recursivos. Secuencia *Interview*.

En la tabla 4.6 se muestran los porcentajes de compensación obtenidos para las tramas 6 y 7. No obstante que el movimiento es de mayor magnitud conforme evoluciona la secuencia, los porcentajes de pixels no compensados no varían considerablemente. Por ejemplo, para BD, varía entre 12.1% y 15.5%. Algo similar ocurre con los otros métodos. Por tanto, los resultados de la tabla dan una idea del comportamiento general de los porcentajes a lo largo de la secuencia.

	<b>BD</b>	<b>WR</b>	<b>BZ1</b>	<b>BZ2</b>
% pixels compensados (recursividad)	83.55	85.45	86.03	84.90
% pixels compensados (detección de mov.)	1.15	1.28	1.28	1.14
% pixels compensados (iteratividad)	2.77	6.83	6.63	7.93
% pixels no compensados	12.51	6.42	6.04	7.93
promedio de iteraciones/pixel	0.15	0.14	0.25	0.25

Tabla 4.6. Porcentajes de compensación obtenidos para los métodos originales (BD, WR) y los combinados (BZ1, BZ2). Secuencia *Interview*.

Resulta de interés observar las imágenes de pixels no compensados. En la figura 4.23 se muestran las correspondientes a la estimación entre las tramas 6 y 7 para los dos métodos originales, mientras que en la figura 4.24 se muestran para comparación las generadas por la estimación entre las tramas 19 y 20. En la figura 4.23 se aprecia nuevamente que los métodos pel-recursivos fallan en la estimación en las fronteras de movimiento (silueta de la mujer en movimiento, rejilla superior del fondo), así como en regiones ruidosas (fondo de la imagen), pero muy especialmente en el pantalón de la mujer (textura muy fina). Si se observa ahora la figura 4.24, se encuentra que las principales regiones no compensadas son del mismo tipo que las del caso anterior, pero en particular no se compensan regiones del brazo y mano izquierdos de la mujer. Este resultado se debe por un lado a la presencia de regiones de oclusión (fondo y objeto en movimiento), pero sobre todo a la magnitud del movimiento, poniendo en evidencia la limitación de los métodos pel-recursivos para desplazamientos muy grandes. Estas imágenes muestran otra vez que la información de pixels no compensados puede ser útil, al menos parcialmente, para operaciones posteriores en las que se requiera de alguna medida de la confiabilidad de la estimación.

#### 4.3.4.2. Calidad de la reconstrucción de la secuencia.

La calidad de la reconstrucción se muestra en la figura 4.25. Los métodos generan resultados satisfactorios (entre 39 y 42 dB). El comportamiento de los métodos es en general similar a lo ya visto en secciones anteriores: el método combinado BZ1 permite la mejor reconstrucción, y el BD, la menos buena (en cuanto a PSNR). Hay que decir que visualmente no se aprecian diferencias entre las secuencias reconstruidas por todos los métodos analizados.

Para finalizar, se presentan para comparación las imágenes de error para el método BD con los dos pares de tramas estudiadas (figura 4.26). Como se esperaba, las regiones donde se presentan los errores mayores son precisamente las regiones donde falla la estimación.

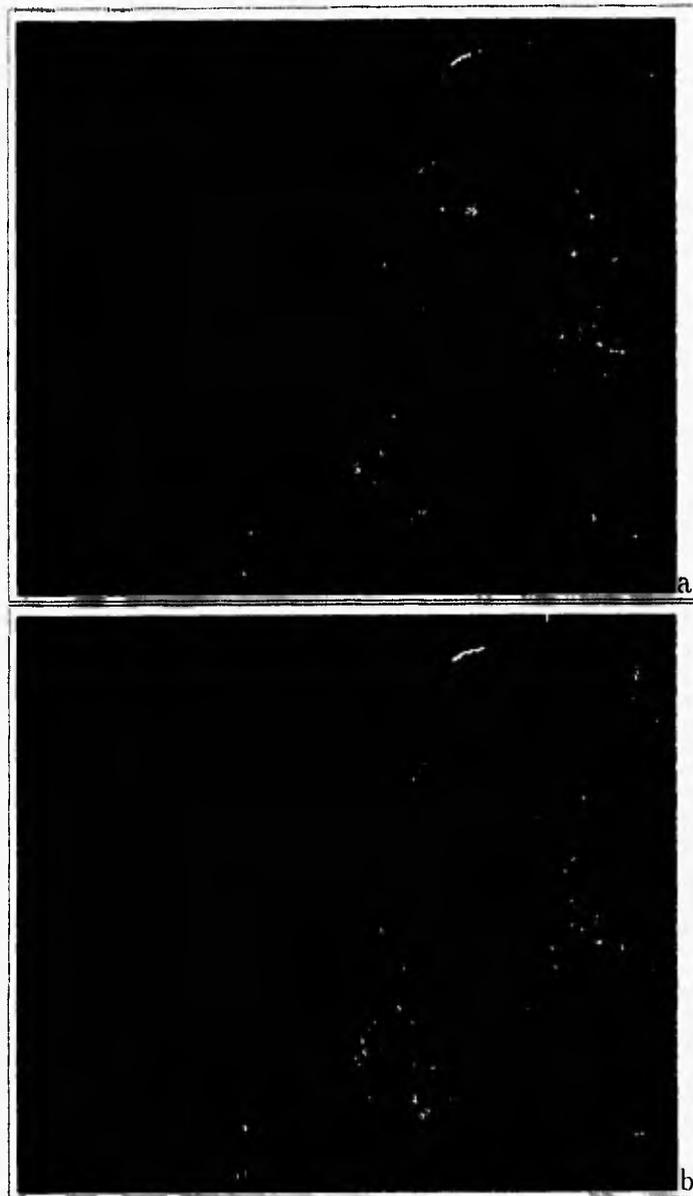


Figura 4.23: Imágenes de pixels no compensados para (a) BD y (b) WR. Secuencia *Interview*. Tramas 6 y 7.

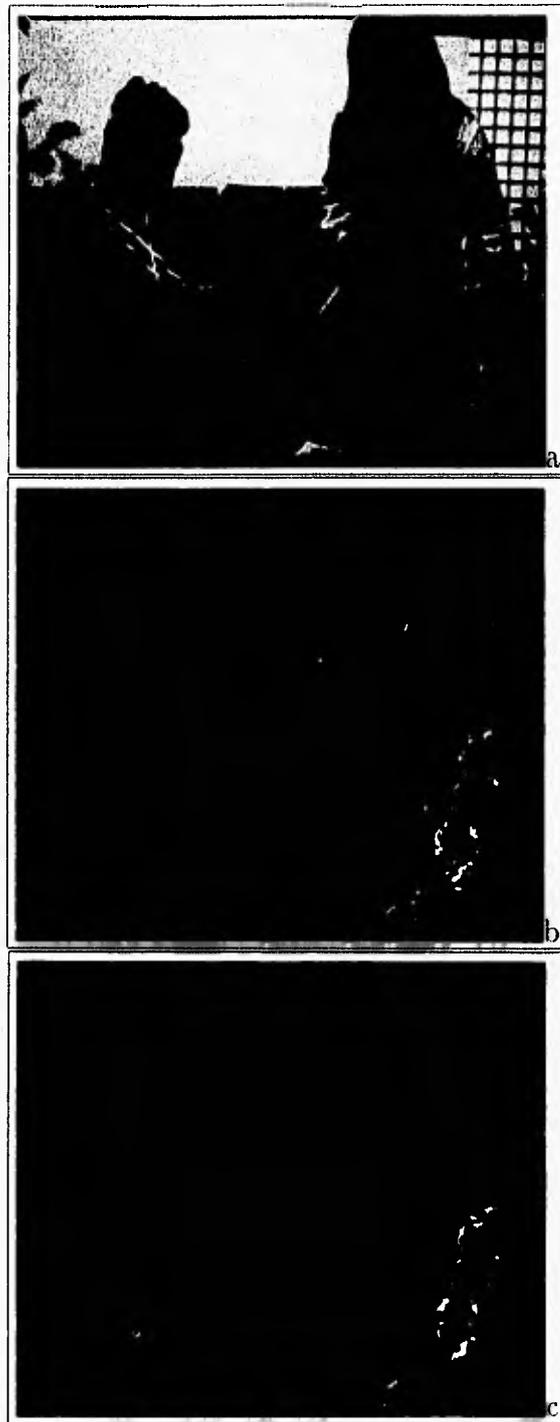


Figura 4.24: (a) Trama 19 de la secuencia *Interview*. Imágenes de pixels no compensados para (b) BD y (c) WR. Estimación entre las tramas 19 y 20.

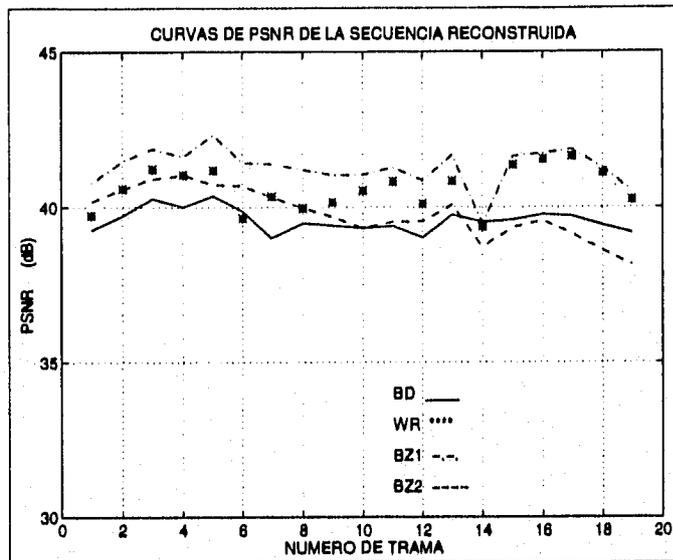


Figura 4.25: PSNR de la secuencia *Interview* reconstruida.

### 4.3.5 Recapitulación

En resumen, para las secuencias sintéticas y reales utilizadas, todos los métodos pel-recursivos mostraron un desempeño satisfactorio en cuanto a calidad de reconstrucción de las secuencias, siendo BD el que presenta resultados numéricos menos buenos (SNR de más de 38.5 dB para las tres secuencias utilizadas). El algoritmo combinado BZ1 fue el que mejor reconstrucción permite, cuantitativamente hablando (SNR superior a 40 dB). Los otros dos métodos producen resultados numéricos intermedios. En cuanto a la apreciación visual de las secuencias reconstruidas, no se notan diferencias significativas entre los cuatro métodos.

Ahora bien, con referencia a la coherencia del campo generado, el método BD es el que produce campos de movimiento más uniformes, menos contaminados por ruido, y más congruentes con la realidad física. En los demás métodos es más notorio el error en la estimación, siendo BZ1 el menos coherente de los campos generados. Este resultado se repite en todas las secuencias, aunque es más evidente en la secuencia *Seqmillt*. Se ha mostrado, en fin, que campos de desplazamiento diferentes producen resultados visuales similares.

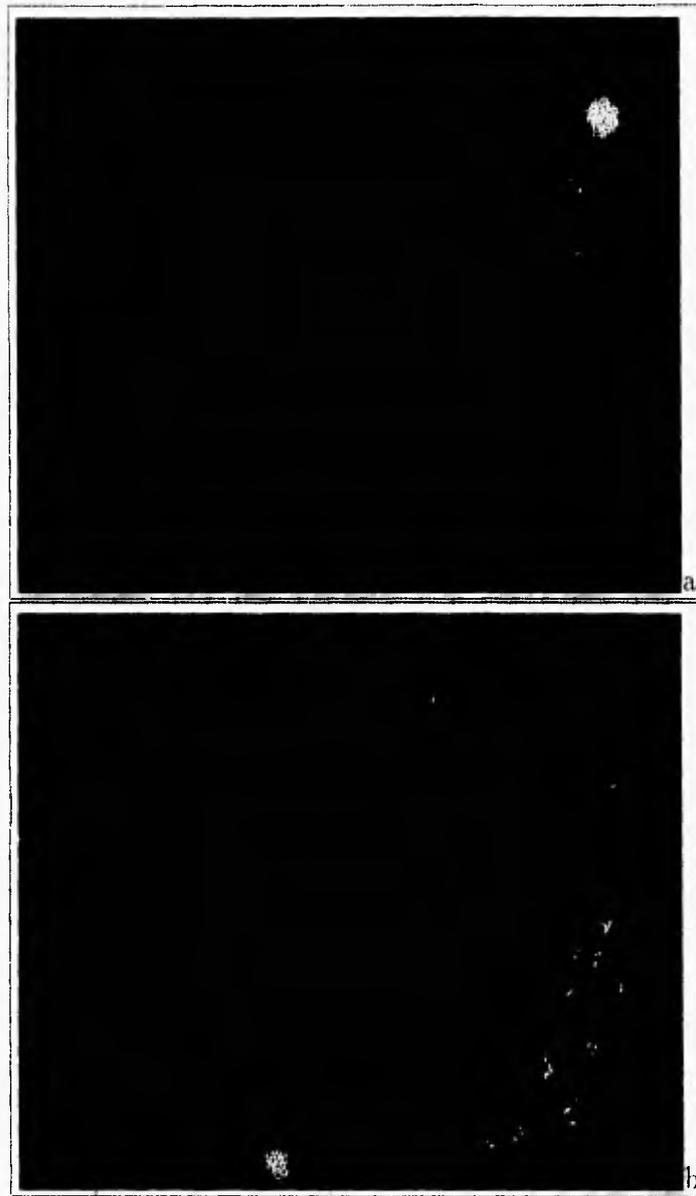


Figura 4.26: Imágenes de error de reconstrucción. Secuencia *Interview*. Método BD. (a) Tramas 6 y 7. (b) Tramas 19 y 20. El error aparece amplificado por 25 para su visualización.

### 4.3.6 Consideraciones para la implantación de los algoritmos.

En esta sección se mencionan algunos aspectos prácticos importantes para la implantación de los algoritmos utilizados, ya que inciden en el desempeño de los métodos. <sup>2</sup>

#### 4.3.6.1. Selección del vector inicial de desplazamiento.

Ya se vio que en un método de optimización local es muy importante la selección del valor o condición inicial a utilizar. Los métodos pel-recursivos originalmente consideraron como valor inicial de desplazamiento  $\hat{\mathbf{d}}^0 = \hat{\mathbf{d}}^{i-1}$  al vector  $\hat{\mathbf{d}}^i$  estimado para el pixel contiguo anterior sobre la misma línea de barrido. La razón es la facilidad computacional que ofrece el no tener que almacenar en memoria el campo vectorial de desplazamiento. Pero es evidente que existen muchas otras posibilidades para  $\hat{\mathbf{d}}^0$ :

1. Un vector perteneciente a alguna vecindad espacial alrededor del pixel analizado [BAAZ91a], [BAAZ91b].
2. Un vector perteneciente a alguna vecindad temporal alrededor del pixel analizado [PAQU83].
3. Combinaciones de 1 y 2.
4. El vector nulo  $\hat{\mathbf{d}}^0 = \mathbf{0}$ .

Estas opciones para la elección de  $\hat{\mathbf{d}}^0$  tienen cualitativamente un respaldo, al menos parcial, pero también presentan problemas [MOOR87]:

1. Si los pixels de la vecindad espacial elegida pertenecen al mismo objeto o región en movimiento, es razonable pensar que aproximadamente experimentarán un desplazamiento similar, es decir, esta opción supone que los vectores de desplazamiento siempre tienen una gran correlación espacial, lo cual no es cierto en las fronteras de movimiento, donde los vectores están altamente descorrelacionados.
2. La elección temporal del  $\hat{\mathbf{d}}^0$  supone que la localización espacial de los objetos en movimiento no varía mucho de trama a trama. Esto es sólo aproximadamente cierto para secuencias con movimiento muy pequeño. Con esta técnica, las estimaciones del desplazamiento en los bordes en movimiento pueden mostrar discontinuidades notables, pero no se resuelve completamente el problema.

---

<sup>2</sup>Por ejemplo, en [BAAZ91b] se emplea una combinación de parámetros distinta a la utilizada aquí, a fin de optimizar los valores de PSNR de reconstrucción, sin tomar en cuenta las características de coherencia de los campos de movimiento.

Históricamente, para los algoritmos pel-recursivos el esquema de predicción espacial se prefirió sobre el esquema de predicción temporal. Una tercera opción propuesta en [MOOR87] se basa en la predicción *a lo largo de la trayectoria del movimiento*, que es por tanto un esquema de predicción espacio-temporal. Es evidente que la solución ideal debe apuntar en esta dirección, hacia un enfoque que combine el suavizado del campo espacial o temporal de manera selectiva. Este razonamiento conduce a la necesidad de una carta de etiquetas (segmentación) de discontinuidades o rupturas de movimiento, para realizar la estimación de manera más precisa. Los métodos más recientes de estimación de movimiento [KONR92], [CHEO93], [DUBO93], [HEIT93], [HUAN93] toman en cuenta este concepto. La segunda parte de este trabajo está dedicada justamente a encontrar dicha segmentación, con lo cual pueda mejorarse la estimación.

El criterio de selección del  $\hat{d}^0$  que se elige para la implantación de los algoritmos en este trabajo es el mismo que el usado en [BAAZ91b]:  $\hat{d}^0$  se hace igual al  $\hat{d}^1$  previamente estimado para algún vecino espacial, de forma que produzca la  $DFD$  más pequeña en valor absoluto para el pixel analizado.

Los métodos de codificación predictiva originalmente empleados con los algoritmos pel-recursivos obligan a la elección de una vecindad causal de tamaño variable para la selección del vector de desplazamiento inicial. Se realizan pruebas con varios tamaños de vecindades espaciales  $N_{vec}$ .

X X	X X X	X X X X	X X X X X	X : pixel vecino
X O	X O	X X O	X X X O	O : pixel actual

$N_{vec} = 4$     $N_{vec} = 5$     $N_{vec} = 7$     $N_{vec} = 9$

El tamaño de la vecindad espacial incide en el desempeño de los métodos, tanto para la calidad del campo como para la reconstrucción de las secuencias. Las gráficas 4.27 y 4.28 ilustran estos resultados, usando la secuencia *Seqmill*, y el método BD. Se observa que, conforme aumenta  $N_{vec}$ , también lo hacen la  $H_d$  y la PSNR. La razón es la mayor libertad otorgada a la condición inicial para poder converger a distintos mínimos locales de la  $DFD$ , que significa una mejor reconstrucción pero que tiende a introducir errores en la estimación del movimiento aparente. Este resultado también se aprecia cuando las pruebas se realizan con las secuencias reales.

#### 4.3.6.2. Tamaño de la vecindad de observación $N$ .

Se dijo en el capítulo 3 que, para el método BD,  $N$  tiene como finalidad producir una estimación más estable y robusta al ruido. El efecto que produce es el de suavizar las características del campo. Si  $N$  es muy pequeño, la restricción de suavizado local impuesta es débil, y el beneficio de la inclusión de la vecindad es

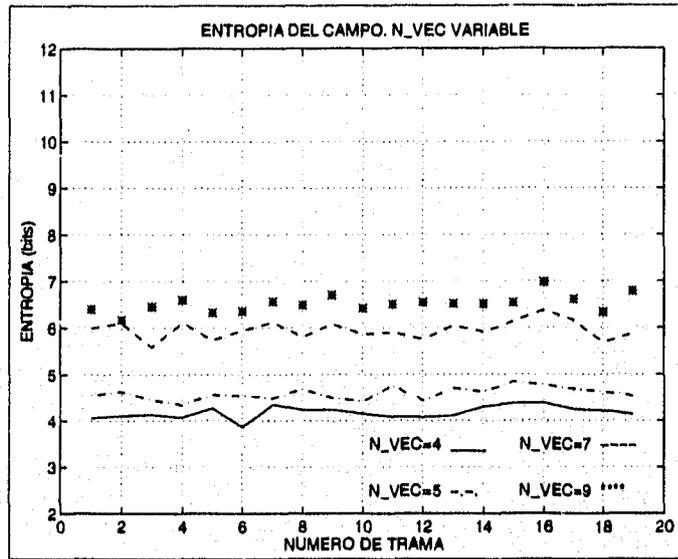


Figura 4.27: Entropía de los campos de desplazamiento estimados. Método BD.  $N_{vec}$  variable. Secuencia *Seqmilt*.

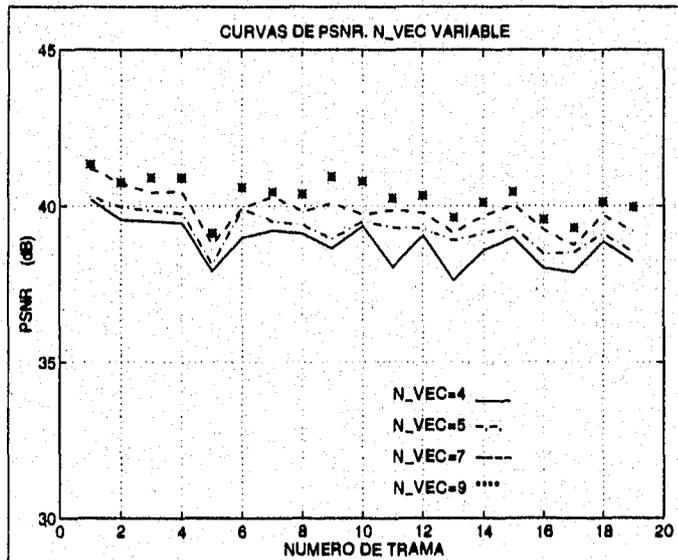


Figura 4.28: PSNR de la secuencia *Seqmilt* reconstruida. Método BD.  $N_{vec}$  variable.

bajo; si la vecindad crece demasiado, el campo puede verse afectado por un suavizado excesivo, sobre todo en las fronteras de movimiento. Se encontró experimentalmente que los valores de  $N = 7$  y  $N = 9$  proporcionan para el método BD los campos más homogéneos.

#### 4.3.6.3. Número de iteraciones por pixel.

En todos los resultados mostrados se efectuó una sola iteración por pixel. El efecto que  $it_{max}$  produce conforme aumenta está orientado siempre hacia la minimización de la  $DFD$ . Por tanto, se espera que el error de reconstrucción disminuya, y que la entropía del campo aumente, al hacerse cada vez más evidente el efecto de optimizar localmente la función de costo. Esto es justo lo que se muestra en las figuras 4.29 y 4.30, para el método BD (secuencia *Seqmilt*). Con 5 iteraciones se obtiene una ganancia de 2 dB aproximadamente en la PSNR, mientras que con 10 iteraciones el incremento en ganancia es marginal. Esto apunta a que la ganancia tiende a estabilizarse después de un cierto número de iteraciones (en [BAAZ91b], por ejemplo, no se usan más de 5). Con respecto a  $H_d$ , ésta se incrementa rápidamente conforme aumenta  $it_{max}$ , en busca de una mejor reconstrucción. Finalmente, este comportamiento se mantiene cuando el algoritmo se aplica a secuencias reales. Se concluye por tanto que un valor de  $it_{max} = 1$  es el ideal para mantener el equilibrio entre buena coherencia del campo y buena reconstrucción, para el método BD.

Para los demás métodos se presenta una tendencia similar. Sin embargo, es importante mencionar que la ganancia en PSNR más grande ocurre en los métodos combinados, aunque perceptualmente no haya cambios.

#### 4.3.6.4. Selección del método de implantación del gradiente.

La diferenciación numérica para el cálculo del gradiente debe realizarse de manera cuidadosa, ya que puede ser una fuente de error considerable [BARR94]. La complicación mayor proviene del hecho de que la derivación es en sí misma un problema mal planteado [BERT88]. Por tanto, para cualquier método diferencial de estimación se espera que el desempeño final muestre alguna dependencia del método numérico de cálculo de los gradientes. Existe además un compromiso entre la precisión deseada en la estimación de las derivadas y la complejidad computacional requerida. En este trabajo se realizaron pruebas con varias configuraciones de diferencias finitas. El operador local que produjo mejores resultados es una modificación del operador para detección de bordes de Prewitt [HARA92] usado también en [BAAZ91a]:

$$\frac{1}{6} \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix} \quad \frac{1}{6} \begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}$$

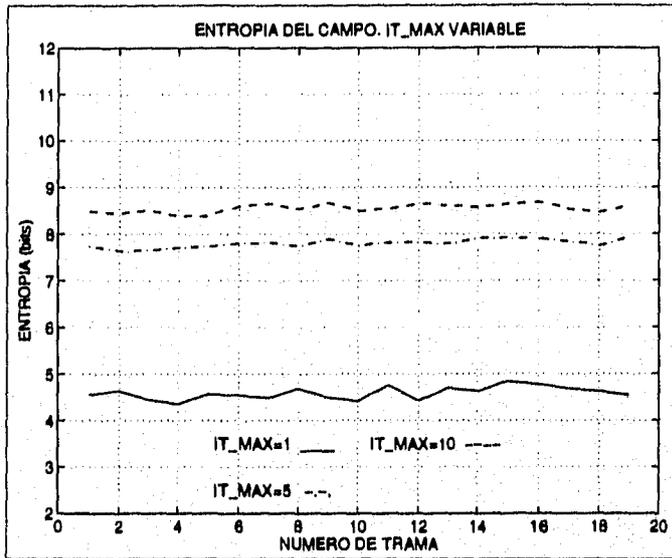


Figura 4.29: Entropía de los campos de desplazamiento estimados. Método BD.  $it_{max}$  variable. Secuencia *Seqmill*.

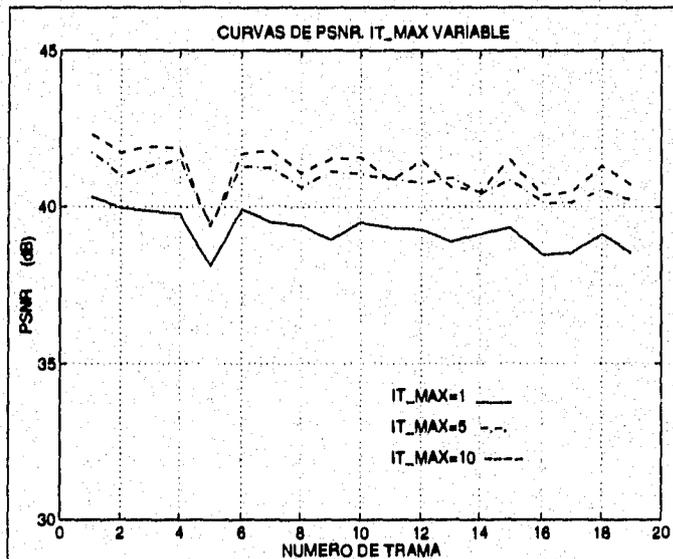


Figura 4.30: PSNR de la secuencia *Seqmill* reconstruida. Método BD.  $it_{max}$  variable.

para el cálculo de las derivadas en  $x$  y  $y$ , respectivamente.

#### 4.3.6.5. Interpolación de la intensidad de la imagen.

Para mantener una buena precisión numérica, el campo de desplazamiento en este trabajo se representa por números flotantes. En el algoritmo, es necesario calcular la  $DFD$  y  $\nabla I$  en la posición desplazada  $(x - d_{xi}, y - d_{yi})$ . Por lo tanto, es necesario realizar operaciones de interpolación de la intensidad de la imagen. En este trabajo se emplea interpolación bilineal en el cálculo de la  $DFD$ , mientras que  $\nabla I$  se calcula con redondeo del vector de desplazamiento [NETR79]. Otros órdenes de interpolación (bicuadrático, bicúbico) se han probado anteriormente [PAQU83], reportándose mejoras modestas en el desempeño a costa de un aumento notable en el costo computacional.

#### 4.3.6.6. Limitación del término de actualización.

Para reducir problemas de convergencia en el método WR, este término se acota entre valores extremos (mínimo  $1/16$  y máximo 2 pixels por iteración). Además, se limita el valor absoluto máximo de las componentes del vector de desplazamiento actualizado  $\hat{d}^i$  a 10 pixels horizontal o verticalmente por trama. En caso de que la estimación exceda estos valores durante el proceso de convergencia, se restablece a cero el valor de  $\hat{d}^i$ . En cambio, el método BD no impone ninguna de estas limitantes.

#### 4.3.6.7. Selección de los umbrales.

Los umbrales  $T_{FD}$  y  $T_{DFD}$  propuestos originalmente, son fijos, determinados de manera heurística y con valores entre 2 y 4. Es evidente que la eficacia que se logra con estos umbrales es dependiente del ruido y de la cantidad de movimiento presentes en cada secuencia [BIER86], [HOTT88].

#### 4.3.6.8. Pixels no compensados.

A estos pixels puede asociárseles cualquier valor de desplazamiento, ya que para ellos no se encuentra un desplazamiento que satisfaga el criterio de convergencia después de realizar el número máximo de iteraciones permitidas. Un criterio que puede adoptarse [BAAZ91b] para minimizar el error de predicción y mejorar la reconstrucción de las imágenes, es elegir aquel desplazamiento (de entre el desplazamiento nulo, el valor inicial  $\hat{d}^0$  asociado al pixel analizado, y el valor final  $\hat{d}^i$  después de la última iteración calculada) que produzca la  $DFD$  más baja.

### 4.3.7 Limitaciones de los métodos.

Se han comentado ya diversos aspectos positivos y negativos de los algoritmos pel-recursivos. A continuación se enlistan sus limitaciones generales.

1. Los métodos son altamente dependientes del valor de predicción o vector de desplazamiento inicial.
2. La obtención de los gradientes no siempre es buena, especialmente en las fronteras de movimiento. Esto, junto con una mala inicialización, ocasiona errores en la estimación en dichas fronteras (y por tanto en las regiones de oclusión). Regularmente es en las discontinuidades de movimiento donde se producen los errores de reconstrucción más grandes.
3. Estas técnicas son sensibles a áreas ambiguas: localmente es imposible determinar el movimiento de una región homogénea, porque los gradientes espaciales son nulos (véase la figura 4.17). En este caso, es la lógica del algoritmo lo que determina el efecto de inercia de movimiento.
4. En zonas altamente texturadas (como en la región del pantalón de la mujer en movimiento en la secuencia *Interview*) las estimaciones suelen ser poco precisas.
5. Debido a la naturaleza recursiva de los algoritmos, los errores en la estimación pueden propagarse en la dirección de barrido.
6. La hipótesis original de la invariancia de la intensidad no se cumple, por ejemplo, en presencia de sombras.
7. Existe un límite en el desplazamiento que pueden estimar con precisión. Los enfoques jerárquicos permiten la estimación de desplazamientos mayores, pero también hasta cierta magnitud.
8. Por sí mismos, estos esquemas de estimación local no proporcionan información de movimiento global o de alto nivel. Pero sí pueden servir como una fuente para la extracción de esta información mediante procesos posteriores, que tomen en cuenta las limitaciones inherentes a la estimación (ver sección siguiente).

Hay que decir, sin embargo, que algunas de estas limitaciones son compartidas por todos los algoritmos de estimación diferenciales, y que otros métodos presentan sus propias limitaciones.

### 4.3.8 Aplicaciones.

Ya se ha apuntado la aplicación típica de estos métodos: la compresión de secuencias de imágenes. Ahora bien, como se puede ver de las gráficas de las entropías  $H_d$ , un campo vectorial denso de movimiento representa una cantidad muy grande de información (dos terminos,  $d_x$  y  $d_y$ , por pixel). El aprovechamiento de la información de movimiento para la reducción del monto total de datos a transmitir tiene entonces dos opciones:

- Utilizar un esquema del tipo DPCM para transmitir solamente una señal de error (sin transmisión de vectores de desplazamiento).
- Transmitir un solo vector que represente todo un conjunto de pixels.

La primera opción ha sido la elegida clásicamente para los algoritmos pel-recursivos [GATI96]. En este caso, se transmite el error de predicción:

$$DFD = I(z, t) - I(z - \hat{d}^{i-1}, t - 1)$$

que en el receptor se utiliza para reconstruir la secuencia. En la figura 4.31. se muestran los valores de entropía del error de predicción obtenidos con las tres secuencias de prueba utilizadas, y se comparan con la entropía de la señal  $FD$ . Como puede verse, los métodos WR, BZ1 y BZ2 generan mejores resultados que BD, aunque en todos los casos se logran compresiones pequeñas (alrededor de 3 para *Seqmilt* y para *Miss América* y de 4 para *Interview*). Para esquemas pel-recursivos más complejos (a múltiples resoluciones como el de [BAAZ91b]) se reportan tasas de compresión de alrededor de 10. Normalmente se logran compresiones mayores (a) al considerar esquemas de cuantificación adicionales, que se ajusten a las características de la señal de error [NETR79], [WALK84], [BIEM87], [WALK87], [BAAZ91b], y (b) al ajustar algunos parámetros en el modelo pel-recursivo (en particular los umbrales  $T_{FD}$  y  $T_{DFD}$ ). En ambos casos, ésto se logra a costa de un cierto nivel de degradación en las secuencias reconstruidas. Sin embargo, este esquema de codificación queda limitado cuando se desean tasas de compresión más altas. Como variantes, recientemente han surgido esquemas [LI93] que se basan en la segmentación de la señal de error de predicción para aumentar la tasa de compresión.

La segunda opción de codificación (un vector de movimiento para un grupo de pixels) fue originalmente usada por los algoritmos de puesta en correspondencia entre bloques. Los últimos años han marcado nuevas tendencias para conseguir compresiones a tasas altas (sin degradar en exceso la calidad visual de las secuencias), aplicando resultados provenientes del análisis de imágenes y de la visión computacional, y que se conocen con el nombre genérico de *métodos de codificación basados en modelos*. En ellos, se representa a las imágenes empleando algún *modelo estructural* que toma en cuenta algunas propiedades de la escena original, y se considera además un *modelo de movimiento* [NICO92b],[LI94], [AIZA95], [GARC95a]. Una

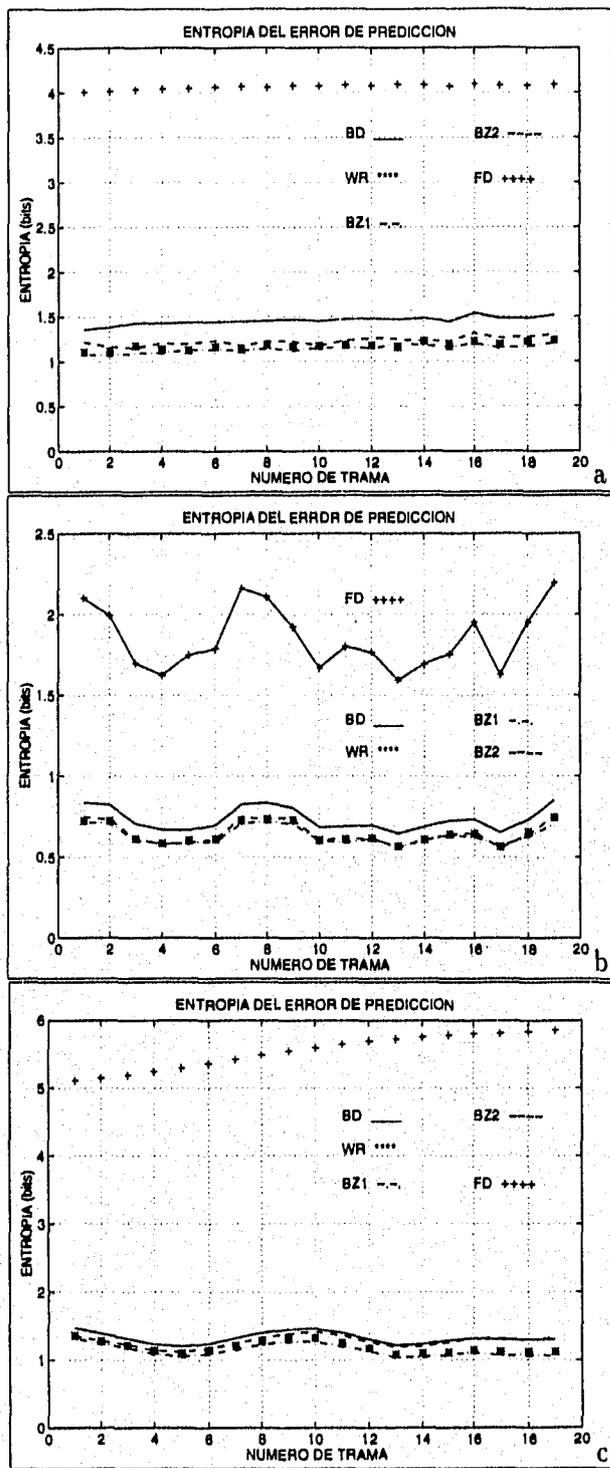


Figura 4.31: Entropía del error de predicción obtenida con los métodos pel-recursivos. (a) Secuencia *Seqmill*. (b) Secuencia *Miss América*. (c) Secuencia *Interview*.

ventaja de estos métodos es que describen el contenido de la imagen de manera estructural, y explotan estas características (contornos, regiones) para la codificación. El concepto clave es el de representar cada región *homogénea* (en algún sentido), por un *conjunto de parámetros* compacto y preciso.

Se ha mencionado la importancia de la información de movimiento para la compresión de secuencias. Por tanto, ésta aparece naturalmente como un criterio conveniente para guiar la partición de la imagen. De esta manera, una tarea importante para muchos métodos de codificación modernos es la de la **segmentación** de la imagen en regiones, **basada en el movimiento aparente** presente en la secuencia.

En realidad, la segmentación basada en movimiento en una secuencia es un problema complejo en sí mismo, y ha dado origen a toda una línea de trabajo dentro del análisis de movimiento. De entre los diferentes enfoques para realizar esta tarea, uno de los más utilizados se basa justamente en la **segmentación de un campo de movimiento aparente previamente estimado**. Esto es, el *agrupamiento* o *fusión* de aquellos vectores de desplazamiento o velocidad que correspondan a un *movimiento homogéneo* o *similar*.

Ahora bien, una segmentación basada en movimiento no sólo encuentra aplicaciones en codificación. En visión computacional, la explotación de la información de movimiento contenida en una segmentación es necesaria en problemas complejos como el *seguimiento de objetos*, o la *interpretación* (cualitativa o cuantitativa) de la escena [FRAN91], [NICO92b]. La importancia del proceso de segmentación se enfatiza en el capítulo siguiente.

Regresando a los métodos analizados en este capítulo, se puede concluir que, a diferencia de los otros algoritmos analizados, el basado en estimación de Wiener (BD) presenta un conjunto de cualidades que lo hacen atractivo como punto de partida para una segmentación basada en movimiento:

- Si bien no es perfecto, el campo estimado es coherente con la realidad física.
- La precisión en la estimación del movimiento es aceptable.
- El campo presenta características de suavizado local, que hacen que la estimación tolere la presencia de ruido [BIEM87]; al mismo tiempo, este suavizado no es tan severo como para eliminar las fronteras de movimiento.
- Como producto adicional a la estimación, proporciona un mapa de pixels no compensados, que pueden utilizarse como información útil para la segmentación.
- El algoritmo mostró un desempeño similar en todas las secuencias de prueba utilizadas.

### **Propuesta de aplicación.**

A partir de lo anteriormente discutido, se propone utilizar el campo de desplazamiento aparente estimado por el método BD como guía para efectuar una segmentación basada en movimiento en una secuencia de imágenes. Partiendo del hecho de que no existe un algoritmo de estimación *perfecto*, dada la naturaleza misma del problema de la medición del movimiento, el algoritmo de segmentación deberá incluir no solamente la información del campo, sino consideraciones adicionales, a fin de aprovechar al máximo la información confiable, y de tratar de paliar algunas de las limitaciones inherentes a la estimación.

### **4.3.9 Conclusiones.**

En este capítulo se presentó un análisis detallado del desempeño de algunos de los métodos pel-recursivos más conocidos. Se mostraron los resultados obtenidos con secuencias sintéticas y reales, considerando como criterios principales de evaluación, la calidad de reconstrucción de las secuencias por compensación de movimiento, y las características de los campos estimados, en cuanto a continuidad y coherencia con la realidad física. Se determinó que todos los métodos permiten una buena reconstrucción, aunque los campos sean bastante diferentes. Como parte final de la evaluación, se discutieron las limitaciones y perspectivas de este enfoque de estimación de movimiento.

Por otra parte, tomando como base la aplicación original de estos métodos (compresión de secuencias), se encontró que los algoritmos pel-recursivos producen por ellos mismos una compresión baja, aunque puede verse mejorada por bloques funcionales adicionales. Finalmente, al considerar las propiedades cualitativas y cuantitativas de los campos estimados, se mostró que el campo generado por el método basado en estimación de Wiener (BD) presenta características que lo hacen aparecer como un buen candidato para ser utilizado como punto de partida para una *segmentación basada en movimiento*, proceso muy importante tanto para codificar secuencias bajo un enfoque orientado a modelos (que permite alcanzar tasas de compresión más altas), como para funciones de visión computacional. Por tanto, se propone su uso para la implantación de esta tarea. Las características particulares del algoritmo de segmentación basada en movimiento se detallan en la segunda parte de este trabajo.

## **Parte II**

# **Segmentación basada en movimiento utilizando campos aleatorios de Markov**

## Capítulo 5

# Segmentación basada en movimiento aparente en secuencias de imágenes.

### 5.1 Introducción.

Se mencionó al inicio de este trabajo que el movimiento aparente extraído de una secuencia de imágenes, en forma de un campo de desplazamiento o velocidad, es una valiosa fuente de información acerca de la naturaleza y las características de los objetos que conforman la escena vista en la secuencia. También se mencionó que un enfoque común para el análisis de movimiento consta de dos fases: la estimación del campo vectorial, y la explotación del mismo en procesos subsecuentes para obtener información de nivel superior. En este sentido, la partición de la escena en regiones con movimiento similar aparece como una tarea relevante.

En general, una *segmentación basada en movimiento (SBM)* puede verse como un proceso de *partición de la imagen en regiones* que tengan movimiento 2D o 3D similar, o como un proceso de *detección de las fronteras de movimiento* entre estas regiones. En particular, una *segmentación de campo de movimiento (SCM)* se define como el *agrupamiento* de aquellos vectores de desplazamiento o velocidad que correspondan a algún *movimiento homogéneo*. Este proceso de fusión de datos (o asignación de etiquetas de segmentación) supone el uso de algún modelo para la descripción del movimiento. Posteriormente, las regiones de pixels definidas por la segmentación pueden asociarse a diferentes partes de un mismo objeto, o a distintos objetos en movimiento dentro de una escena. Los dos problemas (SBM y SCM) son idénticos cuando se dispone de un campo vectorial denso [TEKA95].

En un sistema de visión artificial, la idea de segmentar una escena basándose en el movimiento descansa sobre una importante analogía biológica: el movimiento es una guía poderosa para la segmentación en el sistema visual humano (recuérdense los casos de detección de animales camuflados en movimiento, o de patrones de

puntos superpuestos con movimiento relativo) [POTT75], [MURR87].

### 5.1.1 ¿Para qué segmentar una secuencia de imágenes en base al movimiento?

Dentro del análisis de secuencias de imágenes, la SBM es una tarea básica - y al mismo tiempo compleja - para las siguientes aplicaciones [ADIV85], [HOTT88], [FRAN91], [NICO92], [LI94], [AIZA95], [GARC95b], [TEKA95]:

1. **Mejoramiento en la estimación del campo de movimiento.** Del capítulo anterior, se concluye que la estimación y la segmentación del movimiento en una escena son dos problemas íntimamente relacionados: para una estimación precisa del campo, deben conocerse las fronteras de los objetos en movimiento, ya que de otra forma se introducen errores en la estimación en estas zonas de la imagen (recuérdese por ejemplo las vecindades espaciales  $N$  y  $N_{vec}$  en los algoritmos pel-recursivos); por otra parte, para obtener una segmentación correcta, se necesita de una descripción adecuada de los vectores de desplazamiento (figura 5.1). Este problema - no trivial - se conoce como la *ambigüedad estimación-segmentación* de movimiento. Las alternativas para enfrentar esta situación son tres: (a) realizar primero la estimación y posteriormente la segmentación con métodos separados, (b) invertir el orden de realización de las funciones, o (c) ejecutar ambas tareas simultáneamente (en un mismo algoritmo). Como se definió en el capítulo anterior, la estrategia utilizada en este trabajo es la primera, es decir, realizar la SBM a partir del campo de desplazamiento previamente estimado con el método pel-recursivo BD (basado en Wiener). Pero para cualquier método de estimación, es válido pensar que, partiendo de la interdependencia entre ambos procesos, una SBM puede permitir la identificación de las discontinuidades de movimiento de los objetos y/o de las regiones de oclusión, para ajustar posteriormente la estimación del campo de movimiento.
2. **Descripción de alto nivel del contenido de una escena.** La información generada por la segmentación (orientada a regiones u objetos), puede considerarse como cualitativamente superior, comparada con la información de bajo nivel que representan los vectores de movimiento asociados a cada pixel de la imagen. En particular, en los métodos de codificación de secuencias de imágenes a tasas bajas y muy bajas, el modelado de la imagen por regiones y su segmentación permiten obtener una representación bastante compacta (un vector de parámetros de movimiento por cada región de la escena), que asegura un mínimo de información a transmitir, y una buena calidad de reconstrucción.
3. **Estimación de la estructura y del movimiento 3D de los objetos en una escena.** En aplicaciones como robótica y navegación, es necesario

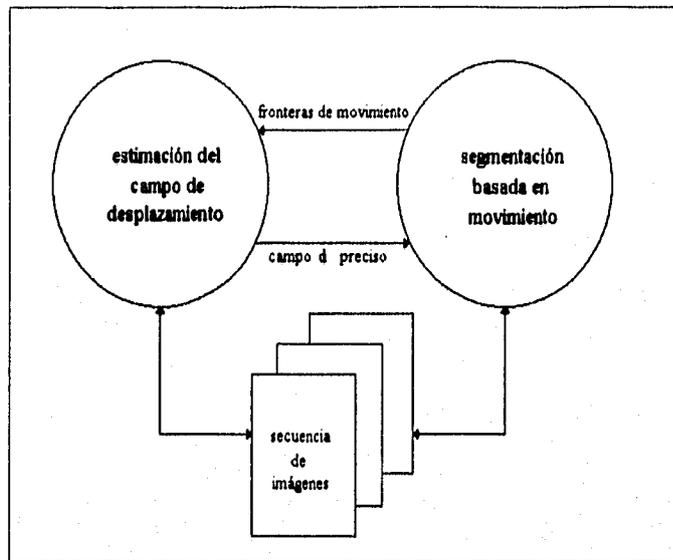


Figura 5.1: Estimación y segmentación de movimiento: dos problemas interdependientes.

encontrar para cada región (objeto), el conjunto de parámetros que describa mejor su movimiento, de acuerdo al modelo elegido.

### 5.1.2 Aspectos particulares de una SBM.

Una SBM, también conocida como *segmentación espacio-temporal*, y una segmentación espacial son dos problemas distintos [FRAN91]. Existen varias diferencias básicas:

- Una región homogénea en cuanto al desplazamiento puede no ser homogénea espacialmente (pueden existir gradientes espaciales en su interior). La situación contraria también puede ocurrir: dos regiones adyacentes con movimiento diferente que no presenten discontinuidades espaciales fuertes entre ellas.
- La información de la que se dispone en ambos casos es fundamentalmente diferente: mientras que en una segmentación espacial se tiene acceso directo a la información a analizar (niveles de gris), en una SBM no ocurre siempre lo mismo: como se verá en la sección siguiente, existen algoritmos de SBM que no requieren del cálculo previo del campo de movimiento. Para estos métodos, en ausencia de restricciones adicionales, sólo se dispone de las variaciones espacio-temporales de la intensidad de la imagen, que permiten obtener sólo una componente de la información necesaria (problema de apertura).

La definición de un *criterio de homogeneidad* en cuanto al movimiento es la base que guía una SBM. Estos criterios evidentemente deben descansar en alguna

medida de similitud entre los vectores de desplazamiento, pero pueden también incluir información adicional, como por ejemplo, acerca de la calidad de las secuencias reconstruidas (esto último es empleado en codificación) [NICO92b]. Es claro entonces que los criterios de homogeneidad para una segmentación espacial no son los mismos que para una SBM, por lo cual la aplicación directa de los métodos simples de agrupamiento espacial a una SBM puede no producir resultados útiles. En general, un objeto con movimiento real produce un campo de desplazamiento no homogéneo desde un punto de vista puramente translacional (en una rotación como en el cuadrado de la secuencia *Seqmilt*, los vectores de movimiento estimados no se podrían asociar) [CHAN93], [TEKA95]. Por ello, cualquier algoritmo para agrupar los vectores en regiones debe tomar en cuenta un modelo de movimiento más elaborado. Con esto, los *parámetros del modelo* se convierten en las características reales a partir de las cuales se lleva a cabo la fusión de pixels en regiones.

## 5.2 Métodos para SBM en una secuencia de imágenes.

El problema de la SBM ha sido abordado desde hace mucho tiempo [POTT75]. Desde entonces, un buen número de algoritmos se ha propuesto (tabla 5.1). Una primera clasificación, en base al rango de aplicaciones, los divide en métodos de *aplicación genérica*, y métodos de *aplicación específica*.

Se puede elaborar una segunda clasificación siguiendo la jerarquía clásica usada en segmentación espacial: *orientada a contornos*, y *orientada a regiones*, pero donde se tienen ahora contornos y regiones espacio-temporales.

En los métodos que buscan extraer contornos, se pueden distinguir los que buscan *contornos en movimiento* (contornos espaciales que se mueven, aún dentro de una región con movimiento homogéneo) y los que buscan *fronteras de movimiento* (discontinuidades de movimiento). Ahora bien, ya se apuntó que la detección de las fronteras de movimiento es indispensable para una estimación correcta del campo de desplazamiento, pero la segmentación que producen estos métodos es insuficiente para la interpretación del mismo, ya que es difícil identificar las diferentes regiones con movimiento similar dentro de la imagen [FRAN91].

Por otro lado, los métodos de SBM orientados a regiones pueden subclasificarse en *métodos directos*, que calculan la segmentación a partir de los gradientes espacio-temporales de la secuencia, e *indirectos*, que la realizan partiendo de un campo de movimiento previamente estimado por alguno de los métodos mencionados en el capítulo 2 de este trabajo. Una mención aparte la merecen aquellos métodos que realizan la estimación y la segmentación *simultáneamente*.

Criterio de clasificación	Clases de métodos	Métodos específicos
Rango de aplicaciones	<ul style="list-style-type: none"> <li>● aplicación genérica</li> <li>● aplicación específica</li> </ul>	
Características estructurales buscadas	<ul style="list-style-type: none"> <li>● orientados a contornos</li> <li>● orientados a regiones</li> </ul>	<ul style="list-style-type: none"> <li>● contornos en movimiento</li> <li>● fronteras de movimiento</li> <li>● directos (sin estimación <math>d</math> previa)</li> <li>● indirectos (campo <math>d</math> previo)</li> <li>● estimación y segmentación simultáneas</li> </ul>

Tabla 5.1. Clasificación de los métodos de segmentación basada en movimiento en secuencias de imágenes.

A continuación, se describen con más detalle los métodos orientados a regiones, ya que producen resultados con interpretación más útil, y porque el método propuesto en este trabajo es de este tipo.

### 5.2.1 Métodos directos de SBM.

El método directo de segmentación más simple es el basado en el cálculo de la  $FD$  sobre una vecindad, y en su comparación con un umbral, de forma que se binarice la imagen en dos tipos de regiones: *móviles* (objetos en movimiento y regiones de oclusión) e *inmóviles* [BIER86]. Este esquema se conoce como *detector de movimiento*. Una situación que se presenta aquí es la asignación errónea de pixels sin movimiento a las regiones móviles y viceversa, debido al ruido presente en las secuencias; además, normalmente las regiones móviles obtenidas son de mayor dimensión que las regiones reales, y quedan algunas regiones móviles pequeñas aisladas. En estos casos, un postprocesamiento adecuado (filtrado de mediana o morfológico) puede mejorar la segmentación final. Es evidente que esta binarización no aporta mucha información, pues contempla en realidad un solo parámetro (existencia o no de movimiento), pero sirve como un primer paso para esquemas de segmentación mucho más elaborados [HOTT88].

Los métodos directos más conocidos son los propuestos por Hotter y Thoma [HOTT88] y Diehl, y pueden definirse como algoritmos *jerárquicamente estructurados de arriba hacia abajo*, porque comienzan con un número muy pequeño de regiones

y conforme el algoritmo avanza, estas regiones se van dividiendo en regiones más pequeñas, dependiendo de qué tan bien se ajuste el modelo de movimiento propuesto a cada región o subregión. En [HOTT88] se utiliza un modelo de ocho parámetros (movimiento 3D rígido planar bajo proyección perspectiva). Originalmente, el detector de movimiento descrito en la sección anterior inicializa la segmentación, asociando cada región móvil espacialmente conectada con un objeto diferente. Una vez hecho esto, se estiman los parámetros del modelo para cada objeto. La validez de los parámetros para cada región se verifica al calcular el error de reconstrucción ( $DFD$ ). En el siguiente nivel de la jerarquía, las regiones de la imagen que no fueron descritas correctamente por el modelo se procesan de la misma forma que al inicio del algoritmo. El procedimiento se repite hasta que el modelo describe de forma adecuada cada región. Las diferencias básicas del método de Diehl están en el modelo de movimiento utilizado (12 parámetros suponiendo una superficie cuadrática) y en el método de estimación de los parámetros [TEKA95].

Un enfoque directo estadístico es el de Bouthemy y Santillana [FRAN91], que se basa en pruebas de verosimilitud siguiendo un *enfoque jerárquico de modelos de movimiento*. En él, la SBM se efectúa usando primero un modelo constante para el campo de velocidad, y después un modelo lineal (6 parámetros), para el refinamiento de las regiones. El procedimiento es del tipo crecimiento de regiones, y el criterio para la fusión de dos regiones en una sola se determina por medio de una razón de verosimilitud. Las dos hipótesis para la prueba de verosimilitud surgen de suponer, por un lado, una sola parametrización para las dos regiones, y por otro, la preservación para cada región de su propio vector de parámetros de movimiento. El orden en el cual se realiza la fusión de regiones se basa en la creación de una tabla de todas las parejas posibles de regiones a fusionar, ordenada de forma tal que la primer pareja de la tabla represente la mejor minimización del criterio, y así sucesivamente. Cada vez que hay una fusión, la tabla se recalcula y reordena, hasta que el valor más pequeño del criterio de fusión sea mayor que un umbral, con lo que termina el algoritmo.

### 5.2.2 Métodos indirectos de SBM.

En todos estos métodos, se cuenta siempre con un campo de movimiento calculado a priori y con un modelo paramétrico asociado al campo (figura 5.2). Los objetivos son dos: encontrar un número  $N_R$  de regiones que describan correctamente el movimiento de cada uno de los objetos de la escena, y determinar el conjunto de parámetros  $\Theta_R$  del modelo para cada región. De esta manera, la estrategia general es asignar cada vector de movimiento a aquella región cuyos parámetros lo representen mejor. Es evidente que al inicio de estos métodos, tanto  $N_R$  como  $\Theta_R$  se desconocen, por lo que los métodos regularmente son iterativos.

Algunos de los enfoques indirectos más representativos son el método basado en transformada de Hough, y los métodos bayesianos basados en campos de Markov.

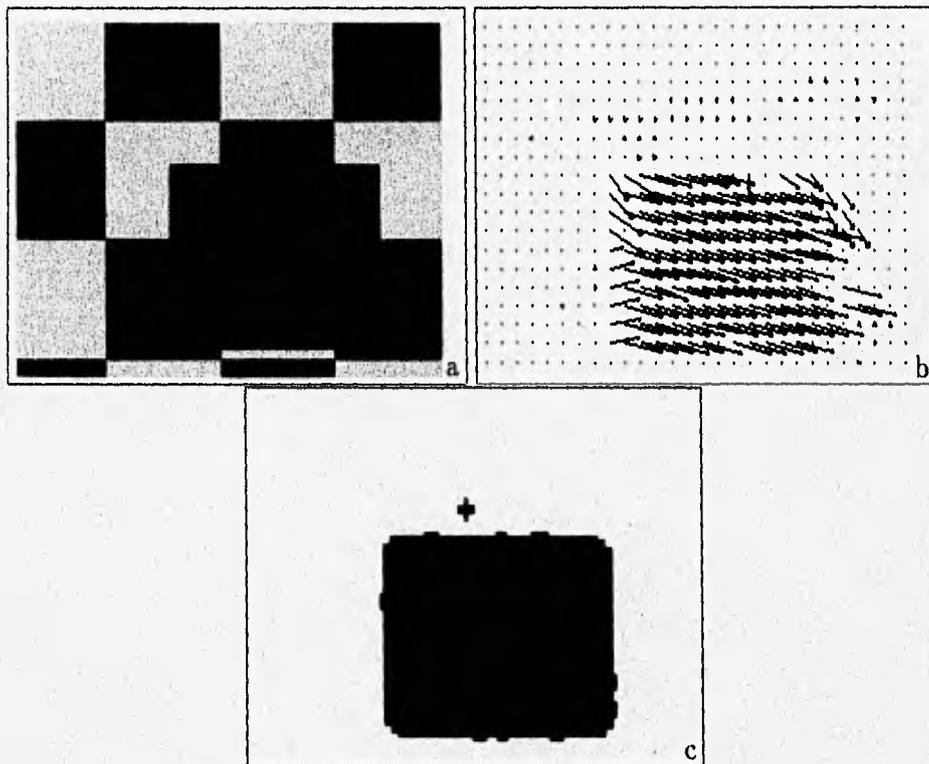


Figura 5.2: El concepto de segmentación basada en movimiento indirecta. A partir de la secuencia de imágenes (a), se estima un campo de movimiento (b), y éste se utiliza como la información que guía la segmentación (c).

Ambas estrategias se describen brevemente en las subsecciones siguientes.

#### **5.2.2.1. Método basado en transformada de Hough.**

En [ADIV85] se presenta un método que emplea un modelo de movimiento 3D rígido planar bajo proyección perspectiva, y que consta de dos etapas de segmentación. En la primera fase, se agrupan los vectores en pequeñas regiones que sean congruentes con una superficie planar en movimiento, de acuerdo a una versión simplificada de la transformada de Hough (HT), utilizada para no elevar demasiado el costo computacional. Entre las variantes propuestas están: (a) la descomposición del modelo original de 8 parámetros en dos subconjuntos ajenos de tres parámetros cada uno (transformación affine), a fin de aplicar posteriormente la HT de tres dimensiones a cada subconjunto, y (b) un esquema de HT a múltiples resoluciones. En la segunda etapa del método, las regiones generadas por la primera fase se reagrupan bajo el supuesto de que algunas de ellas corresponden a un solo objeto rígido con movimiento 3D, por medio de un procedimiento de *inspección de hipótesis (hypothesis testing)*. Cada hipótesis se prueba buscando los parámetros 3D (basándose en una técnica de mínimos cuadrados) que sean compatibles con todos los segmentos en el grupo correspondiente. En realidad, pueden formularse distintos criterios *ad hoc* para la fusión de regiones. En una fase final, los vectores finales que no fueron agrupados son asimilados por alguna de las regiones vecinas, quedando entonces determinados los parámetros de movimiento 3D de toda la escena.

#### **5.2.2.2. Métodos bayesianos.**

Una SBM puede plantearse y resolverse como un problema de *estimación bayesiana*, que busque globalmente la probabilidad a posteriori máxima del campo de *etiquetas de segmentación* dado el campo de desplazamiento (criterio *máximo a posteriori MAP*). Un modelado utilizando campos aleatorios de Markov permite incorporar, en el mismo esquema y de manera directa, conocimientos previos de naturaleza múltiple acerca del problema, en forma de interacciones espacio-temporales locales, con el fin de obtener en el resultado final las características deseadas para la segmentación. Una vez definido el modelo, la segmentación bayesiana se convierte en un problema de optimización global que se resuelve mediante cálculos locales.

Murray y Buxton fueron los primeros en usar este enfoque para SBM [MURR87]. Las *observaciones* (campo de velocidad) se modelan bajo la suposición de movimiento 3D de superficies planas, y se usa una función de densidad de probabilidad de Gibbs para integrar el conocimiento a priori acerca de la segmentación. El método requiere del establecimiento previo del número de regiones distintas dentro de la imagen, ya que conmuta iterativamente entre la estimación de los parámetros del modelo de movimiento para cada región, y la optimización del campo de etiquetas, hasta alcanzar la convergencia. La probabilidad a maximizar en este caso es

$$p(\mathbf{z}/\mathbf{v}_x, \mathbf{v}_y) = p(\mathbf{v}_x, \mathbf{v}_y/\mathbf{z})p(\mathbf{z})/p(\mathbf{v}_x, \mathbf{v}_y) \quad (5.1)$$

donde  $\mathbf{z}$  es el campo de etiquetas de segmentación, y  $(\mathbf{v}_x, \mathbf{v}_y)$  es el campo de velocidad. El algoritmo introduce, al igual que el método pionero de los Geman [GEMA84], un campo de *líneas de discontinuidad*, considerando similarmente sólo *cliques* de primer orden. La maximización de la probabilidad a posteriori se realiza por la técnica de *recocido simulado*, por lo que el método resulta sumamente costoso computacionalmente, ya que además de la relajación estocástica, se requiere de la estimación de los parámetros de cada región por mínimos cuadrados.

Un segundo enfoque se encuentra en [FRAN91], que se utiliza como la primera fase de un sistema de interpretación cualitativa de movimiento en secuencias de imágenes. Este método se basa en el tratamiento estadístico hecho por Bouthemy y Santillana, pero con una modelización markoviana, por lo que en realidad, se trata de un método directo de SBM. Aquí se parte de la definición del campo de *sitios* como bloques pequeños de pixels, lo cual limita la precisión en la segmentación, pero asegura tolerancia al ruido y menor número de cálculos. La función a maximizar es

$$p(\nabla I, I, \mathbf{z}, \theta, N_R) \quad (5.2)$$

donde  $\theta$  es el vector de parámetros del modelo,  $\mathbf{z}$  es el campo de etiquetas de segmentación, y  $N_R$  es el número de regiones. Dentro de la función de energía a minimizar se introducen las características de homogeneidad deseadas. El esquema de relajación es de tipo determinístico, variante de los métodos ICM (*iterated conditional modes*), y utiliza una tabla de *etiquetas de inestabilidad*.

Un tercer enfoque markoviano aparece en [CHAN93], que plantea una estrategia de segmentación bayesiana que incluye información de nivel de gris de la imagen, suponiendo que el campo de desplazamiento  $(\mathbf{d}_x, \mathbf{d}_y)$  y la intensidad de la secuencia  $\mathbf{i}$  son estadísticamente independientes, modelando la probabilidad como

$$p(\mathbf{z}/\mathbf{d}_x, \mathbf{d}_y, \mathbf{i}) = p(\mathbf{d}_x/\mathbf{z})p(\mathbf{d}_y/\mathbf{z})p(\mathbf{i}/\mathbf{z})p(\mathbf{z})/p(\mathbf{d}_x, \mathbf{d}_y, \mathbf{i}) \quad (5.3)$$

La información de nivel de gris se utiliza cuando las estimaciones para el desplazamiento no son confiables (de acuerdo al valor de la *DFD*). De los dos casos presentados en [CHAN93] para el modelado de la probabilidad condicional  $p(\mathbf{d}_x, \mathbf{d}_y/\mathbf{z})$ , presenta mejor desempeño el basado en modelos paramétricos, por lo que este algoritmo en realidad representa una modificación del algoritmo de Murray y Buxton.

### 5.2.2.3. Métodos simultáneos de estimación y segmentación.

Los métodos simultáneos como el de [HEIT93] o [TEKA95] requieren de la introducción de gran cantidad de información, en vista de los dos procesos que deben realizar; de aquí que hayan elegido a los MRF como su esquema de modelado. El primero de los métodos entrega un campo de velocidades junto con un campo de líneas de discontinuidad de movimiento (SBM orientada a contornos), mientras que el segundo proporciona junto con el campo de velocidad, un campo de etiquetas de segmentación y un vector de parámetros por cada región obtenida. La optimización del problema se realiza utilizando variaciones de los métodos ICM, a fin de reducir la complejidad computacional.

## 5.3 Discusión.

De los métodos comentados, ¿cuál es el mejor? Esto no se puede responder de manera única, porque las características y aplicaciones potenciales de cada método son diferentes. Pero, pensando en su desempeño, y en posibles aplicaciones y modificaciones, puede mencionarse lo siguiente:

1. Todos los métodos utilizan modelos paramétricos de movimiento.
2. Los métodos directos tienen la ventaja de que no necesitan del campo de desplazamiento, pero las soluciones ofrecidas son bastante *ad hoc*. Por ejemplo, en los esquemas propuestos por la escuela alemana (Hotter, Thoma, Diehl), los algoritmos se centran en una determinada aplicación y tratan de explotar al máximo el contenido del tipo específico de secuencias (videoconferencia). Ya que el conocimiento a priori sobre el tipo de secuencia a procesar es considerable, se limita su rango de aplicaciones potenciales. En estos esquemas rígidos, la manera de introducir otro tipo de información en los algoritmos no es clara.
3. Por otra parte, si bien la formulación de la solución ofrecida por Bouthemy y Santillana es acertada, sus limitaciones (observadas en secuencias reales) provienen esencialmente del proceso particular de crecimiento de regiones. Por un lado, existe una cierta inestabilidad temporal, ya que muchas regiones desaparecen, aparecen o se dividen de una trama a otra. Además, este enfoque tiende a una segmentación excesiva (sobre todo en las zonas con poca textura). En realidad, estos problemas fueron la motivación para su replanteamiento desde un punto de vista markoviano por François.
4. El método de Adiv fue uno de los primeros en considerar movimiento de varias superficies al mismo tiempo, y funciona adecuadamente. Hay que decir, sin embargo, que consta de un número considerable de etapas, y que los criterios para la fusión de las pequeñas regiones originales en segmentos mayores siguen siendo arbitrarias.

5. Un campo de movimiento siempre contiene errores en la estimación; por tanto, la eficacia de cualquier método indirecto depende de la precisión del campo previamente estimado. Pero como ya se mencionó, un proceso iterativo de estimación y segmentación mejora ambas tareas.
6. Los métodos de SBM bayesianos ofrecen la flexibilidad que no se observa en otros casos. Permiten, por un lado, incorporar en un mismo modelo información variada de manera natural, y al mismo tiempo mantienen a los esquemas suficientemente generales como para ser aplicados a una variedad de secuencias. Como se sabe, los MRF representan una técnica confiable que se ha aplicado con éxito a un amplio número de problemas en procesamiento de imágenes y en visión artificial, como segmentación espacial, restauración, reducción de ruido, reconstrucción de superficies, estéreo visión, y detección y estimación de movimiento [GEMA84], [MARR85], [KONR92], [PAPP92], [CHEL93], [HEIT93], [LI95], [TEKA95], por lo que su empleo resulta aconsejable. Por último, los métodos de relajación determinísticos y la alta capacidad de paralelización de los algoritmos, los hacen cada vez más atractivos.
7. Los métodos simultáneos de estimación-segmentación son elegantes y proporcionan excelentes resultados en cuanto a la estimación final del campo de movimiento, pero son excesivamente complejos. Por ejemplo, en [HEIT93] se requieren entre 200 y 400 iteraciones del método para obtener convergencia, mientras que en [TEKA95] se presentan resultados para 100 iteraciones (y eso considerando estrategias de relajación determinísticas). En vista de esta dificultad, una combinación de dos métodos independientes: uno de estimación y otro de segmentación puede ofrecer buenos dividendos con una complejidad menor.

Considerando los puntos anteriores, y buscando un método que combine algunas de las mejores características de algoritmos anteriores, se propone en el capítulo siguiente un esquema de SBM bayesiano indirecto basado en un modelo paramétrico, que toma como base el campo de desplazamiento generado por el método pel-recursivo basado en Wiener (BD). El esquema propuesto puede verse como una modificación al algoritmo de [MURR87], pero con rasgos propios, que surgen de un planteamiento más general del problema de SBM con una estrategia markoviana.

## Capítulo 6

# Método propuesto de segmentación basada en movimiento.

### 6.1 Introducción.

Como se dijo, una SBM puede plantearse como un problema de *estimación bayesiana global*, que permita la extracción de la información de la segmentación, en forma de un *campo de etiquetas* que asocia cada pixel de la imagen a una determinada región en movimiento (representada por un vector de parámetros). Las interacciones entre la información previa disponible y la segmentación se especifican utilizando *campos aleatorios de Markov (MRF)*, que introducen las restricciones deseadas para la solución y permiten la fusión de datos. Ya se apuntó que los algoritmos basados en un enfoque markoviano se pueden desarrollar a partir de hipótesis bastante generales - y no *ad hoc* como otras técnicas - y permiten asociar información contextual a priori de naturaleza diversa.

### 6.2 Planteamiento general del problema.

#### 6.2.1 Campos de Markov y distribución de Gibbs.

Sea  $\mathbf{e} = \{e_s, s \in S\}$  un conjunto de variables aleatorias (denominado *campo de etiquetas*), definidas en una rejilla bidimensional  $S = \{s_1, s_2, \dots, s_N\}$ . Cada  $s \in S$  se conoce como un *sitio*. Además, sean  $L$  el conjunto de todos los valores posibles de las variables  $e_s$ ,  $e$  una *observación* cualquiera del campo  $\mathbf{e}$ :  $e = (e_{s_1}, \dots, e_{s_N})$ , y  $\Psi$  el conjunto de todas las configuraciones posibles para  $e$ :  $\Psi = \{e = (e_{s_1}, \dots, e_{s_N})\}$ .

Considérese ahora un *sistema de vecindad*  $\mathcal{V}$  definido en  $S$ :  $\{\mathcal{V} = \nu_s, s \in S\}$ , que satisface las propiedades

i)  $s$  no pertenece a  $\nu_s$  (un sitio no pertenece a su propio sistema de vecinos).

ii)  $s_i \in \nu_{s_j} \leftrightarrow s_j \in \nu_{s_i}$  (si  $s_i$  es vecino de  $s_j$ , entonces  $s_j$  debe ser vecino de  $s_i$  y viceversa).

Con esto,  $\mathbf{e}$  es un *campo aleatorio de Markov* para el sistema de vecindad  $\nu$  si

$$i) \forall \mathbf{e} \in \Psi, \quad p(\mathbf{e} = \mathbf{e}) > 0.$$

$$ii) \forall s \in S, \quad p(e_{s_i}/e_{s_j}, \forall s_i \neq s_j) = p(e_{s_i}/e_{s_j}, s_j \in \nu_{s_i}).$$

La última condición significa que la etiqueta en un sitio no depende más que del estado de las etiquetas de sus vecinos (contexto local del enfoque markoviano). Si bien la especificación de un MRF en términos de las densidades de probabilidad condicionales locales es muy complicada, la existencia del teorema de Hammersley y Clifford, que establece la equivalencia entre un campo de Markov y un campo cuya densidad de probabilidad es una *distribución de Gibbs*, facilita la solución del problema. Con ésto, la densidad de probabilidad del campo  $\mathbf{e}$  se puede escribir como

$$p(\mathbf{e} = \mathbf{e}) = \frac{1}{Z} e^{-U(\mathbf{e})/T} \quad (6.1)$$

donde  $T$  es un parámetro conocido como *temperatura*,  $Z$  es una constante de normalización conocida como *función de partición*:

$$Z = \sum_{\mathbf{e} \in \Psi} e^{-U(\mathbf{e})/T} \quad (6.2)$$

y  $U(\mathbf{e})$  es una función conocida como *función de energía* que se puede descomponer como una suma de funciones locales  $V_c$ :

$$U(\mathbf{e}) = \sum_{c \in C} V_c(\mathbf{e}) \quad (6.3)$$

donde  $C$  representa al conjunto de *cliques*  $c$  de  $S$  correspondientes al sistema de vecindad  $\nu$ , y un clique es un subconjunto de  $S$  donde todos los sitios son vecinos mutuos.

Las funciones  $V_c(\mathbf{e})$  son llamadas *funciones de potencial*, y dependen únicamente de los valores  $e_s$ , tales que  $s \in c$ . Es a través de estas funciones que se pueden introducir las propiedades locales deseadas para el campo de etiquetas  $\mathbf{e}$  de acuerdo a las características del problema particular, ya que las  $V_c(\mathbf{e})$  expresan la interacción entre pixels vecinos.

## 6.2.2 Estimación bayesiana global usando MRF.

Un proceso de estimación bayesiana puede entenderse genéricamente de esta forma:

1. Existen por un lado diversas *fuentes* de información que se usan como *observaciones* en el proceso de estimación, como son el desplazamiento aparente estimado, la intensidad de las imágenes (niveles de gris), los gradientes espaciales fuertes, o los pixels no compensados por el método de estimación de movimiento.
2. Las observaciones se combinan de acuerdo a *modelos estructurales locales*, en los cuales se introduce el conocimiento *a priori* que se tiene acerca de las *variables desconocidas (etiquetas)*, de forma tal que éstas se puedan estimar. Para la SBM, dichas variables son las etiquetas de *segmentación* asociadas a cada pixel, pero puede existir otro tipo de información que también resulte útil conocer, como las *discontinuidades* o *fronteras de movimiento* o las *regiones de oclusión* de la imagen.

En resumen, se desea obtener un conjunto de etiquetas a partir de un conjunto de variables observadas. Ahora bien, si se modelan las observaciones y las etiquetas por dos campos aleatorios  $\mathbf{o}$  y  $\mathbf{e}$  que cumplan con lo establecido en la sección anterior, el criterio *máximo a posteriori (MAP)* busca estimar el campo  $\mathbf{e}$  que maximice la densidad de probabilidad global condicional a posteriori:

$$p(\mathbf{e} = e / \mathbf{o} = o) \quad (6.4)$$

es decir desea encontrar

$$\hat{\mathbf{e}} = \operatorname{argmax}_{\mathbf{e}} p(\mathbf{e} = e / \mathbf{o} = o) \quad (6.5)$$

Utilizando el teorema de Bayes, se puede reescribir:

$$p(\mathbf{e} = e / \mathbf{o} = o) = \frac{p(\mathbf{o} = o / \mathbf{e} = e) p(\mathbf{e} = e)}{p(\mathbf{o} = o)} \quad (6.6)$$

Como la maximización no depende del denominador, el criterio MAP se reduce a maximizar la densidad de probabilidad conjunta de las variables observadas y buscadas.

$$p(\mathbf{o} = o / \mathbf{e} = e) p(\mathbf{e} = e) = p(\mathbf{e} = e, \mathbf{o} = o) \quad (6.7)$$

### 6.2.3 Aplicación de la teoría a una SBM.

En el caso que nos ocupa, se puede expresar el problema de una segmentación basada en movimiento como

$$(\hat{\mathbf{z}}, \hat{\mathbf{l}}, \hat{\mathbf{o}}) = \operatorname{argmax}_{\mathbf{z}, \mathbf{l}, \mathbf{o}} p(\mathbf{z}, \mathbf{l}, \mathbf{o} / \mathbf{d}_x, \mathbf{d}_y, \mathbf{i}, \mathbf{p}, \mathbf{g}) \quad (6.8)$$

donde

- $\mathbf{z}$  es el campo de *etiquetas de segmentación*. Esta es realmente la información buscada. Este campo tiene asociado un modelo de movimiento  $\Theta_M$ , y está definido para todos los pixels de la imagen  $S$ . Por tanto, junto con  $\mathbf{z}$ , se encuentra un vector de parámetros  $\Theta$  para cada región.
- $\mathbf{l}$  es un campo de *líneas de discontinuidad de movimiento*. Este es un campo auxiliar que se introduce para mejorar el proceso de segmentación, ya que marca la posición de todas las discontinuidades en el campo de movimiento. Los sitios  $s_l$  de  $\mathbf{l}$  se localizan entre cada par de sitios pixel en el plano imagen, tanto vertical como horizontalmente.  $S_l$  designa el conjunto de todos los  $s_l$  (figura 6.1). Hay dos posibles valores para los elementos del campo de líneas: 1 ó 0, dependiendo si hay o no discontinuidad en la posición respectiva. El estado real de cada sitio del campo de líneas es una variable desconocida, que se estima junto con el campo  $\mathbf{z}$ . El campo  $\mathbf{l}$  se ha usado en otras aplicaciones de MRF [GEM84], [KONR92], [HEIT93], [BRA195a], [GARC95a], y se puede considerar como un resultado secundario del esquema de estimación bastante útil.
- $\mathbf{o}$  es un campo de *oclusión*, definido también sobre  $S$ , que se puede introducir para marcar las regiones del mismo nombre. Cada elemento de este campo puede tomar distintos valores, correspondientes a distintos estados de oclusión (recubrimiento, descubrimiento, no oclusión) [DUBO93].
- $\mathbf{d}_x$  y  $\mathbf{d}_y$  son los campos observados de las *componentes horizontales y verticales del desplazamiento*. Si el campo se estima con un algoritmo pel-recursivo, existe un vector  $\mathbf{d}$  para cada sitio de la imagen  $s \in S$ .
- $\mathbf{i}$  es el campo de *intensidad de la imagen* (niveles de gris), y por tanto está definido en  $S$ . Aunque ya se dijo que esta información por sí misma no es suficiente para clasificar a las regiones de acuerdo a su movimiento, resulta útil cuando la información de la estimación del desplazamiento es imprecisa, como por ejemplo en las fronteras de movimiento.
- $\mathbf{p}$  es el campo binario de *pixels no compensados* por el estimador pel-recursivo, definido sobre  $S$ . Un  $p_s$  vale 0 cuando el pixel en el sitio  $s$  es compensado, y 1 cuando no lo es. Como se vio en la primera parte de este trabajo, los errores

en la estimación ocurren principalmente en regiones ruidosas de la imagen o en las fronteras de movimiento. Por lo tanto,  $p$  se usa para reducir errores en el algoritmo de SBM en estas regiones debido al campo  $d$  mal estimado.

- $g$  es el campo de *bordes espaciales horizontales y verticales* de la imagen, que proporciona información binaria acerca de las discontinuidades de intensidad (1 si hay gradiente fuerte, 0 en caso contrario). Está definido, al igual que el campo de líneas  $l$ , sobre  $S_l$ , y en el modelo se utiliza como evidencia para determinar la existencia de discontinuidades de movimiento.

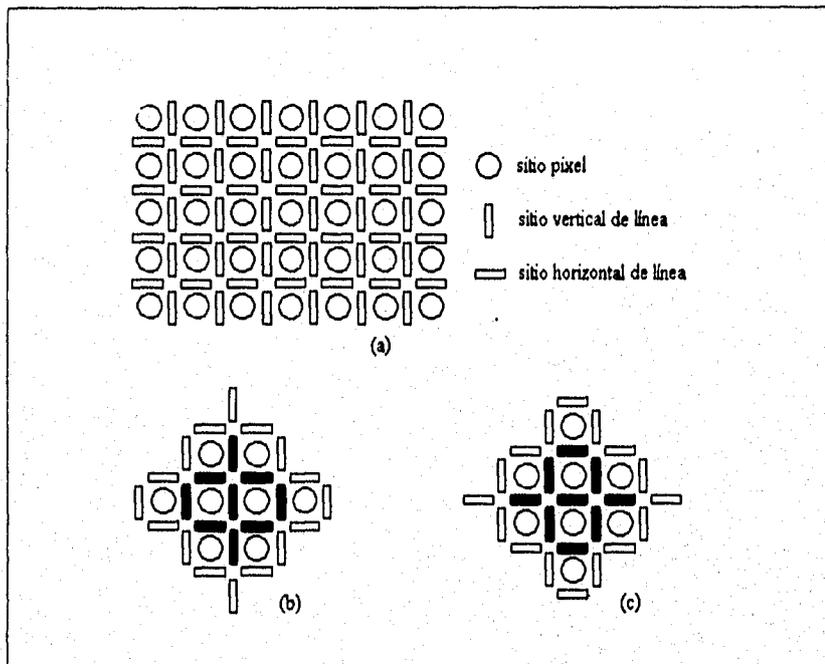


Figura 6.1: (a) Ilustración de las posiciones de los sitios pixel y de los sitios del campo de líneas. Sistema de vecindad de segundo orden del campo de líneas: (b) vertical; (c) horizontal.

A partir de un modelo como éste, conocido como modelo de *campos de Markov acoplados*, pueden desprenderse algunos métodos reportados en la literatura:

- Si se incluyen en el modelo  $z$ ,  $l$ ,  $d_x$  y  $d_y$  se establece el modelo de [MURR87].
- Si sólo se consideran  $z$ ,  $i$ ,  $d_x$  y  $d_y$  dentro del modelo, y se asume independencia estadística entre las observaciones, se obtiene el esquema propuesto en [CHAN93].

El algoritmo propuesto en este trabajo utiliza todas las variables, exceptuando al campo de ocusión  $o$ , por lo cual

$$(\hat{\mathbf{z}}, \hat{\mathbf{l}}) = \operatorname{argmax}_{\mathbf{z}, \mathbf{l}} p(\mathbf{z}, \mathbf{l} | \mathbf{d}_x, \mathbf{d}_y, \mathbf{i}, \mathbf{p}, \mathbf{g}) \quad (6.9)$$

que por teorema de Bayes se puede expresar como

$$(\hat{\mathbf{z}}, \hat{\mathbf{l}}) = \operatorname{argmax}_{\mathbf{z}, \mathbf{l}} p(\mathbf{d}_x, \mathbf{d}_y, \mathbf{i} | \mathbf{z}, \mathbf{l}, \mathbf{p}, \mathbf{g}) p(\mathbf{z} | \mathbf{l}, \mathbf{p}, \mathbf{g}) p(\mathbf{l} | \mathbf{p}, \mathbf{g}) \quad (6.10)$$

ya que el denominador en el cociente de la expresión de Bayes no depende de las etiquetas. Así, el criterio MAP busca maximizar la densidad de probabilidad anterior, que equivale a maximizar la densidad de probabilidad conjunta de observaciones y etiquetas:

$$\max p(\mathbf{z}, \mathbf{l}, \mathbf{d}_x, \mathbf{d}_y, \mathbf{i}, \mathbf{p}, \mathbf{g}) \quad (6.11)$$

En el modelo propuesto en este trabajo se supone independencia estadística entre  $\mathbf{d}$  e  $\mathbf{i}$ , y se omite su dependencia con respecto a  $\mathbf{l}$  y  $\mathbf{g}$ . Además, se establece que el modelo a priori de las etiquetas  $\mathbf{z}$  sólo depende de  $\mathbf{l}$ , y que el modelo a priori del campo de líneas sólo depende de  $\mathbf{g}$ . Estas suposiciones se hacen para que las interacciones entre las distintas variables reflejen adecuadamente el conocimiento que se tiene del problema, integrando todos los aspectos deseados (figura 6.2). Con esto, se desea maximizar

$$(\hat{\mathbf{z}}, \hat{\mathbf{l}}) = \operatorname{argmax}_{\mathbf{z}, \mathbf{l}} p(\mathbf{d}_x, \mathbf{d}_y | \mathbf{z}, \mathbf{p}) p(\mathbf{i} | \mathbf{z}, \mathbf{p}) p(\mathbf{z} | \mathbf{l}) p(\mathbf{l} | \mathbf{g}) \quad (6.12)$$

Si se modelan las dos primeras densidades de probabilidad -correspondientes a las observaciones- suponiendo campos aleatorios gaussianos condicionalmente independientes, y las dos segundas -correspondientes a las etiquetas- por medio de una distribución de Gibbs [GEMA84], [CHEL93] se demuestra que maximizar la densidad de probabilidad anterior equivale a minimizar una función de energía

$$U(\mathbf{z}, \mathbf{l}, \mathbf{d}_x, \mathbf{d}_y, \mathbf{i}, \mathbf{p}, \mathbf{g})$$

que tiene la forma específica

$$U(\mathbf{z}, \mathbf{l}, \mathbf{d}_x, \mathbf{d}_y, \mathbf{i}, \mathbf{p}, \mathbf{g}) = \alpha U_d(\mathbf{d}_x, \mathbf{d}_y, \mathbf{z}, \mathbf{p}) + \beta U_i(\mathbf{i}, \mathbf{z}, \mathbf{p}) + \gamma U_z(\mathbf{z}, \mathbf{l}) + \kappa U_l(\mathbf{l}, \mathbf{g}) \quad (6.13)$$

donde cada término de energía  $U$  se obtiene de la densidad de probabilidad correspondiente en la ecuación (6.12), y  $\alpha$ ,  $\beta$ ,  $\gamma$  y  $\kappa$  son constantes de ponderación. Finalmente, se busca

$$(\hat{z}, \hat{l}) = \operatorname{argmin}_{z,l} U(z, l, d_x, d_y, i, p, g) \quad (6.14)$$

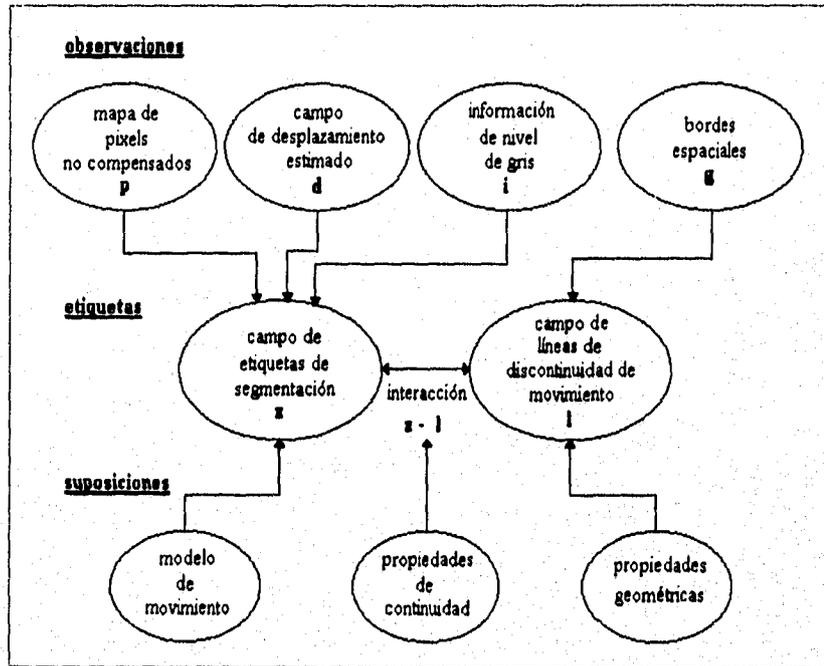


Figura 6.2: Modelo de interacción entre información observada y deseada para el algoritmo de segmentación basada en movimiento.

### 6.3 Modelado de las funciones de energía.

En la ecuación (6.13), los dos primeros términos del segundo miembro,  $U_d$  y  $U_l$ , modelan la relación entre las observaciones y las etiquetas de segmentación (densidad de probabilidad condicional  $p(o/e)$ ), mientras que los otros dos términos,  $U_z$  y  $U_l$ , representan el modelo a priori de las etiquetas (densidad de probabilidad  $p(e)$ ). En las dos secciones siguientes se desarrollan expresiones para cada uno de los términos de energía.

## 6.3.1 Modelo de las observaciones.

### 6.3.1.1. Campo de desplazamiento y pixels no compensados.

La energía  $U_d(\mathbf{d}_x, \mathbf{d}_y, \mathbf{z}, \mathbf{p})$  modela las observaciones del campo de movimiento dado el campo de etiquetas de segmentación y el de pixels no compensados. Por lo tanto, tiene que ser un indicador de qué tan bien el modelo de movimiento  $\Theta_M$  asociado refleja al campo real estimado  $\mathbf{d}$ . El modelo  $\Theta_M$  elegido para este trabajo es el modelo lineal simplificado de 4 parámetros  $\Theta_{MLS}$  (ecuación 2.7) que ha demostrado su equilibrio entre compacidad y complejidad [NICO91], [NICO92b], [GARC95a].

Despréciese por un momento la influencia del campo  $\mathbf{p}$ . Si se supone que el modelo de movimiento propuesto  $\Theta_{MLS}$  es adecuado, entonces de manera similar a [MURR87], la distancia euclidiana entre los vectores de desplazamiento estimados  $\mathbf{d}$  y los sintetizados por  $\Theta_{MLS}$ ,  $\mathbf{d}_\Theta$

$$\delta_{\mathbf{d}, \mathbf{d}_\Theta} = \|\mathbf{d} - \mathbf{d}_\Theta\| \quad (6.15)$$

se puede modelar como ruido gaussiano con media nula y variancia  $\sigma^2$  fija, e independiente en cada uno de los sitios  $s$ . De esta manera, la función de densidad de probabilidad  $p(\mathbf{d}_x, \mathbf{d}_y/\mathbf{z})$  se puede expresar como

$$p(\mathbf{d}_x, \mathbf{d}_y/\mathbf{z}) = \frac{1}{(2\pi\sigma^2)^{N/2}} \exp\left(-\sum_{s=1}^N (\delta_{\mathbf{d}, \mathbf{d}_\Theta})_s^2 / 2\sigma^2\right) \quad (6.16)$$

que es igual a

$$p(\mathbf{d}_x, \mathbf{d}_y/\mathbf{z}) = \exp\left(-\sum_{s=1}^N \left\{ \frac{(\delta_{\mathbf{d}, \mathbf{d}_\Theta})_s^2}{2\sigma^2} + \ln((2\pi)^{1/2}\sigma) \right\}\right) \quad (6.17)$$

Maximizar  $p(\mathbf{d}_x, \mathbf{d}_y/\mathbf{z})$  equivale a minimizar el argumento de la función exponencial. Como el segundo término dentro de la sumatoria es constante, la función de energía a minimizar puede expresarse como

$$U_d(\mathbf{d}_x, \mathbf{d}_y, \mathbf{z}) = \lambda \sum_{s=1}^N (\delta_{\mathbf{d}, \mathbf{d}_\Theta})_s^2 \quad (6.18)$$

El valor de esta energía depende entonces del ruido presente en el campo de desplazamiento, de los errores en la segmentación, y de los errores en el modelado. La energía será menor entre menor sea el error  $\delta_{\mathbf{d}, \mathbf{d}_\Theta}$ .

Considerando ahora al campo  $\mathbf{p}$ , éste proporciona alguna medida de la calidad de la estimación del desplazamiento (al menos, según un criterio de reconstrucción). Para los pixels no compensados, la estimación no es confiable, y por tanto debe reducirse su contribución a la energía total a fin de disminuir efectos indeseables en la segmentación. Para estos casos, hay que darle más valor a otro tipo de información (niveles de gris). Con esto, la función de energía  $U_d(\mathbf{d}_x, \mathbf{d}_y, \mathbf{z}, \mathbf{p})$  adopta la expresión

$$U_d(\mathbf{d}_x, \mathbf{d}_y, \mathbf{z}, \mathbf{p}) = \lambda \sum_{s=1}^N (1 - p_s) (\delta_{\mathbf{d}, \mathbf{d}_0})_s^2 \quad (6.19)$$

### 6.3.1.2. Información de nivel de gris y pixels no compensados.

Cuando la información de movimiento es inexacta (regiones ruidosas o de oclusión), es adecuado hacer que prevalezca el criterio de coherencia espacial local: en una vecindad, pixels pertenecientes a una misma región espacial tienen valores de intensidad similar. Si se modela la diferencia entre la intensidad  $i$  en cada pixel  $s$  y la intensidad media sobre una ventana pequeña  $v$  centrada en  $s$ , como ruido blanco gaussiano [PAPP92], [CHAN93], se encuentra, por un procedimiento similar al realizado en la sección anterior:

$$U_1(\mathbf{i}, \mathbf{z}, \mathbf{p}) = \lambda_2 \sum_{s=1}^N p_s (i_s - i_M(\mathbf{z}))^2 \quad (6.20)$$

donde la interacción con el campo de etiquetas  $\mathbf{z}$  se introduce en el valor de intensidad media  $i_M(\mathbf{z})$  sobre  $v$ , al considerar para el cálculo sólo aquellos vecinos que compartan la misma etiqueta de segmentación que el pixel central  $s$ . La dependencia del campo  $\mathbf{p}$  se manifiesta en el término  $p_s$ : cuando la información de movimiento no sea confiable ( $p_s = 1$ ), se debe enfatizar la coherencia espacial, que proporciona la condición de energía mínima para  $U_1$ .

## 6.3.2 Modelo a priori de la segmentación.

### 6.3.2.1. Etiquetas de segmentación y líneas de discontinuidad de movimiento.

$U_z$  representa el modelo a priori del campo de etiquetas de segmentación  $\mathbf{z}$ , a través de su interacción con el campo de líneas de discontinuidad de movimiento  $\mathbf{l}$  (representado por una distribución de Gibbs):

$$p(\mathbf{z}/\mathbf{l}) = \frac{1}{Z} e^{-U_z(\mathbf{z}, \mathbf{l})} \quad (6.21)$$

Por lo tanto, en su definición deben introducirse las condiciones locales deseadas para que produzcan resultados que correspondan con la realidad física. Esto es, debe favorecerse la *continuidad* en la segmentación (regiones conexas) siempre y cuando las etiquetas de sitios vecinos  $z_{s_1}, z_{s_2}$  pertenezcan realmente a la misma región en movimiento. Para esto último, debe considerarse el hecho de que las líneas de discontinuidad de movimiento marcan las fronteras entre regiones con movimiento distinto. Una función de energía que cumple con estas condiciones es

$$U_z(\mathbf{z}, \mathbf{l}) = \sum_{s=1}^N \sum_{s_j \in \nu_s} (1 - 2\delta(z_s - z_{s_j}))(1 - l(z_s, z_{s_j})) \quad (6.22)$$

donde  $\nu_s$  representa a los cuatro vecinos  $s_j$  más cercanos al sitio  $s$  (conectividad-4),  $l(z_s, z_{s_j})$  es el valor del elemento de línea entre  $z_s$  y  $z_{s_j}$ , y  $\delta$  es el operador de Kronecker. Por tanto, se consideran solamente cliques  $c_z$  de dos pixels (figura 6.3).

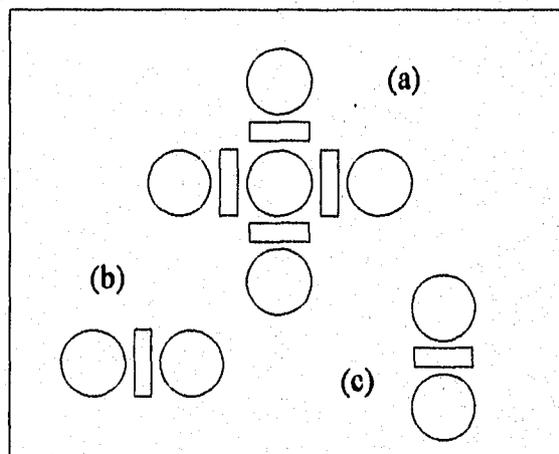


Figura 6.3: (a) Sistema de vecindad de primer orden del campo de sitios pixel (conectividad-4); (b) clique horizontal; (c) clique vertical.

La función de energía  $U_z(\mathbf{z}, \mathbf{l})$  se explica al observar que existe un costo asociado a cada clique  $c_z$  que es mayor si el campo de etiquetas se aleja localmente del modelo a priori propuesto:

- El término  $(1 - 2\delta(z_s - z_{s_j}))$  favorece la continuidad en  $\mathbf{z}$ , ya que  $U_z$  es mínimo (valor de energía negativo) cuando  $z_s$  y  $z_{s_j}$  son iguales, y no hay líneas de discontinuidad presentes entre los pixels ( $l(z_s, z_{s_j}) = 0$ ).
- Por otra parte, si el elemento de línea vale 1, el costo asociado al clique es cero. La finalidad es no introducir penalización por la existencia de etiquetas de segmentación distintas en pixels vecinos, ya que esto es justamente lo que debe ocurrir en las fronteras de movimiento.

- El caso menos favorecido es el que corresponde a incompatibilidades (pixels vecinos con etiquetas de segmentación diferentes, sin línea de discontinuidad entre ellos).

### 6.3.2.2. Líneas de discontinuidad de movimiento y gradientes espaciales.

La capacidad de minimizar la energía  $U_z$ , mediante la inserción o no de líneas de discontinuidad, debe ser penalizada, ya que de otra manera, se podría obtener, por ejemplo, una  $U_z$  total nula si todos los elementos del campo  $l$  valieran 1, que no corresponde a una situación real. Esta penalización se introduce a través del modelo de la energía  $U_1(l, g)$  asociada al campo de líneas, dado el campo de bordes espaciales, que también se modela por una distribución de Gibbs:

$$p(l/g) = \frac{1}{Z} e^{-U_1(l, g)} \quad (6.23)$$

En primer lugar, el modelo a priori de  $l$  debe estar condicionado por las observaciones, en este caso, por la información de gradientes espaciales. En general, una escena que da origen a una discontinuidad de movimiento en ciertos pixels mostrará bordes espaciales en esas mismas posiciones. Por tanto, se supone en el modelo que la introducción de un elemento de línea debe coincidir con un gradiente espacial. En segundo término, en  $U_1$  debe modelarse la *geometría* de las fronteras de movimiento, esto es, debe favorecerse la presencia de ciertas estructuras (líneas rectas, ángulos y esquinas) y penalizar la creación de otras (cruces, líneas dobles, triples, pixels aislados por líneas). La expresión adoptada es igual a la usada en [GARC95a]:

$$U_1(l, g) = \sum_{c_l \in C_l} (l_{s_l}(1 - g_{s_l}) + V_c^l) \quad (6.24)$$

donde  $C_l$  denota el conjunto de todos los cliques para el campo  $l$ ,  $V_c^l$  es el potencial asociado a cada clique, y  $g_{s_l}$  y  $l_{s_l}$  representan los valores de  $g$  y  $l$  en los sitios  $s_l$ .

Para este estudio se considera un sistema de vecindad de líneas de segundo orden, pero sólo se consideran los cliques centrales, donde cada elemento de línea (vertical u horizontal) tiene ocho líneas y dos pixels como vecinos (figura 6.1(b) y (c)). También se incluyen cliques de primer orden formados por 4 líneas [KONR92]. Las configuraciones de cliques y sus valores de potencial son ligeramente diferentes a los usados en [GARC95a] (apéndice 1).

## 6.4 Esquema de optimización.

La minimización global de  $U(\mathbf{z}, \mathbf{l}, \mathbf{d}_x, \mathbf{d}_y, \mathbf{i}, \mathbf{p}, \mathbf{g})$  es un problema bastante complejo, por el gran número de estados posibles del sistema. Las dos estrategias generales de *relajación* utilizadas para resolverlo son las técnicas *estocásticas* y las *determinísticas*.

El primer enfoque (*recocido simulado*) asegura en teoría la convergencia hacia el mínimo global de la energía, pero es excesivamente costoso en cuanto a tiempo de cálculo [GEMA84], [MARR82].

Por otra parte, la relajación determinística permite reducir el tiempo de cálculo, pero no asegura la convergencia al mínimo global, por lo que a los métodos de este tipo se les conoce como *subóptimos*. En estos casos, por lo tanto, es conveniente escoger una buena inicialización, para converger a un mínimo local que represente una solución satisfactoria [FRAN91].

Uno de los métodos determinísticos más conocidos es el ICM (*Iterated Conditional Modes*). Este tipo de métodos se ha usado ampliamente para segmentación de imágenes fijas [PAPP92], y para estimación de movimiento [HEIT93], entre otras aplicaciones. En esencia, consiste en maximizar la probabilidad local

$$p(o_s/e_s)p(e_s/e_s, s_j \in \nu_s)$$

que equivale a minimizar la energía local  $\Delta U_s(e_s)$  en cada sitio  $s$ . En este trabajo se utiliza un esquema de relajación ICM modificado, presentado en [FRAN91].

La minimización de la energía  $U(\mathbf{z}, \mathbf{l}, \mathbf{d}_x, \mathbf{d}_y, \mathbf{i}, \mathbf{p}, \mathbf{g})$  se realiza en dos fases:

1. Optimización del campo  $\mathbf{z}$ , a través de la minimización por ICM del término

$$U_1 = \alpha U_d(\mathbf{d}_x, \mathbf{d}_y, \mathbf{z}, \mathbf{p}) + \beta U_l(\mathbf{i}, \mathbf{z}, \mathbf{p}) + \gamma U_z(\mathbf{z}, \mathbf{l}) \quad (6.25)$$

que corresponde a los primeros tres términos de  $U(\mathbf{z}, \mathbf{l}, \mathbf{d}_x, \mathbf{d}_y, \mathbf{i}, \mathbf{p}, \mathbf{g})$ .

2. Optimización del campo  $\mathbf{l}$ , a través de la minimización por ICM de

$$U_2 = \gamma U_z(\mathbf{z}, \mathbf{l}) + \kappa U_l(\mathbf{l}, \mathbf{g}) \quad (6.26)$$

que corresponde a los últimos dos términos de  $U(\mathbf{z}, \mathbf{l}, \mathbf{d}_x, \mathbf{d}_y, \mathbf{i}, \mathbf{p}, \mathbf{g})$ .

El término  $U_z(\mathbf{z}, \mathbf{l})$  representa el *término de enlace* entre las dos etapas del proceso general de optimización.

La figura 6.4. muestra el diagrama de flujo de la solución propuesta. A continuación se enfatizan algunos aspectos relevantes.

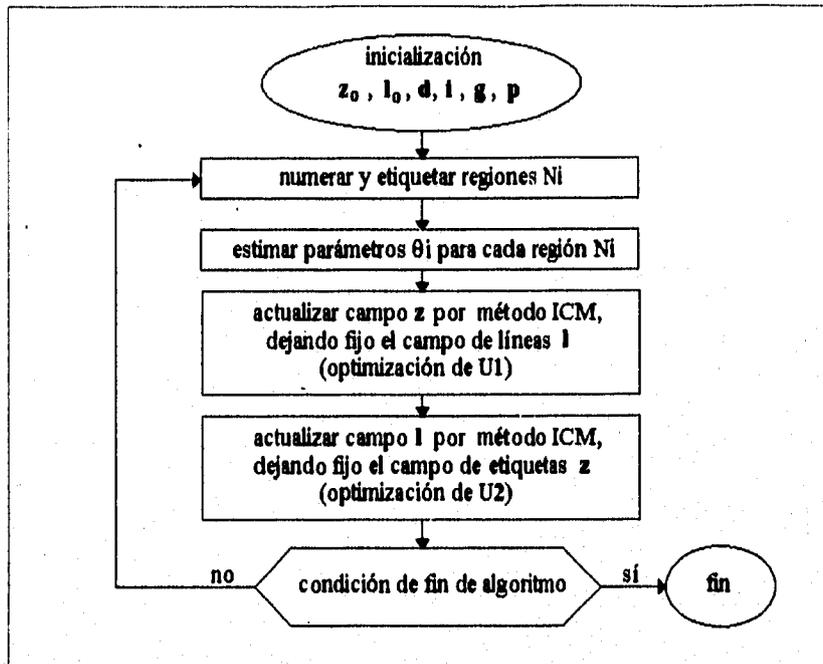


Figura 6.4: Diagrama de flujo del algoritmo de segmentación basada en movimiento propuesto.

### 6.4.1 Inicialización.

Como ya se dijo, un algoritmo ICM requiere de una inicialización adecuada para las variables a optimizar, a fin de converger a un buen mínimo local. Originalmente se dispone de los campos  $l$ ,  $d$  y  $p$  (éstos dos generados por el algoritmo pel-recursivo basado en Wiener). Los demás campos iniciales se definen como:

- $g$ . Se obtiene por un algoritmo de detección de bordes de Mallat.
- $l_0$ . Una primera opción es hacer  $l_0 = g$ . Otra alternativa es partir de las fronteras entre las regiones de la segmentación inicial  $z_0$ .
- $z_0$ . Para este campo, puede elegirse una segmentación cualquiera; sin embargo, es preferible inicializar el proceso de relajación correctamente, y no de manera aleatoria ya que (a) la estimación de los parámetros del modelo de movimiento  $\Theta_M$  requiere de una partición de la imagen desde el inicio, y (b) como el algoritmo es subóptimo, se corre el riesgo de obtener un resultado que no corresponda a la realidad física. En este trabajo, se obtiene a partir de un *detector de movimiento* similar al mencionado en la sección 5.2.1.

Esta inicialización es sólo necesaria para la primera segmentación de una secuencia; para las siguientes, podrían usarse los campos finales  $l$  y  $z$  calculados para la SBM anterior. Esta condición es conveniente para mantener el lazo espacio temporal entre segmentaciones sucesivas y para acelerar la convergencia.

### 6.4.2 Etiquetado y numerado de regiones.

El algoritmo calcula el número de regiones conectadas, y las etiqueta como regiones con movimiento distinto. Al término de cada relajación (etiquetas de segmentación y líneas), se realiza una reenumeración y retiquetado de las  $R$  regiones existentes, ya que la relajación puede llevar a la partición de una región en dos o más subregiones.

Conviene definir un conjunto de *etiquetas de segmentación provisionales*, para aquellos pixels para los cuales existen dudas acerca de su pertenencia o no a alguna región vecina. Esto permite teóricamente la aparición de nuevas regiones en la segmentación. El criterio para el etiquetado de regiones es:

- Si el tamaño de una región es mayor a 30 pixels, se considera como una nueva región válida, y se etiqueta convenientemente.
- Si es menor, se procede a marcarlas como etiquetas provisionales, para que en relajaciones posteriores adopten la etiqueta de segmentación de la región vecina que minimice la energía en esos sitios.

Hay que decir que existen otras formas de tratar a las etiquetas provisionales [FRAN91], [GARC95a].

### 6.4.3 Estimación de parámetros.

Una vez que el algoritmo determina el número de regiones válidas, se calcula el vector de parámetros  $\Theta_i$  para cada región  $R_i$ , con el modelo  $\Theta_{MLS}$ .

### 6.4.4 Esquema de relajación ICM modificado.

Se mencionó antes que la minimización de las funciones de energía se basa en un método determinístico ICM. Ya que la solución obtenida con estos métodos depende del orden en el que se visitan los sitios pixel  $s$  y los sitios de línea  $s_l$  [HEIT93], es preferible utilizar una estrategia de *visita aleatoria* (a fin de evitar efectos direccionales o estructuras parásitas en la imagen de segmentación). En [FRAN91] se propone un método de visita aleatoria que utiliza una *tabla de inestabilidad*. En la figura 6.5 se muestra el diagrama de flujo del método de relajación. El porcentaje máximo final permitido de sitios inestables es una medida de la exigencia que se desea imponer a la optimización, y se usa para detener el proceso. La misma estrategia se utiliza para la actualización de  $z$  y de  $l$ , con ciertas diferencias, porque en  $l$  se tienen elementos verticales y horizontales.

### 6.4.5 Criterio de finalización del algoritmo.

El algoritmo se mantiene en el lazo principal de la figura 6.4 hasta que (a) se llega al número máximo de iteraciones deseadas, o (b) se alcanza la convergencia (de

acuerdo al porcentaje de etiquetas de segmentación o de líneas que cambian entre una iteración y otra).

## 6.5 Diferencias entre el algoritmo de Murray y Buxton y el algoritmo propuesto.

Para finalizar, se presentan en la tabla 6.1, las diferencias principales existentes entre el algoritmo de SBM propuesto en [MURR87] y el algoritmo modificado presentado en este capítulo.

	Murray y Buxton	Algoritmo modificado
<b>Modelo de movimiento</b>	velocidad vernier	$\Theta_{MLS}$
<b>Campo p</b>	no utilizado	sí utilizado
<b>Campo g</b>	no utilizado	sí utilizado
<b>Campo i</b>	no utilizado	sí utilizado
<b>Campo de movimiento utilizado</b>	componente de $v$ normal a los contornos	$d$ estimado por método pel-recursivo BD
<b>Número de regiones</b>	fijo	variable
<b>Segmentación inicial</b>	aleatoria	detector de movimiento
<b>Método de relajación</b>	recocido simulado	ICM modificado

Tabla 6.1. Diferencias entre el algoritmo de [MURR87], y el algoritmo modificado.

## 6.6 Conclusiones.

Se presentó el enfoque propuesto de segmentación basada en movimiento, comentando en detalle tanto los distintos aspectos del modelado markoviano del problema, como la estrategia específica de optimización de la solución.

En el capítulo siguiente se presentan los resultados obtenidos al implantar el algoritmo en computadora, evaluándolo con secuencias sintéticas y reales.

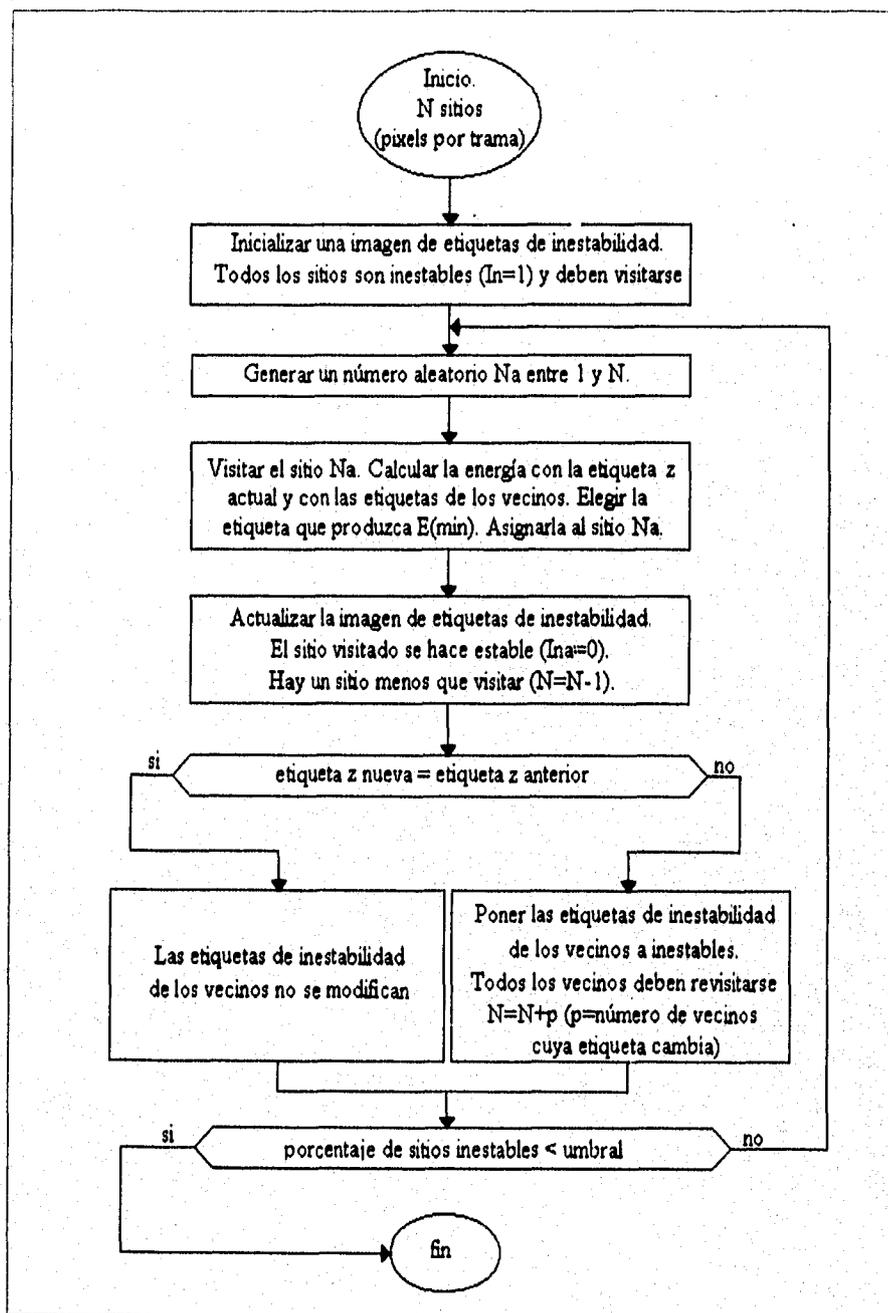


Figura 6.5: Diagrama de flujo del método de relajación determinístico.

## Capítulo 7

# Resultados y evaluación del algoritmo de segmentación basada en movimiento.

### 7.1 Selección de los parámetros del modelo.

La elección de los parámetros de ponderación del modelo markoviano ( $\alpha$ ,  $\beta$ ,  $\gamma$  y  $\kappa$ ) en la ecuación (6.13)

$$U(\mathbf{z}, \mathbf{l}, \mathbf{d}_x, \mathbf{d}_y, \mathbf{i}, \mathbf{p}, \mathbf{g}) = \alpha U_d(\mathbf{d}_x, \mathbf{d}_y, \mathbf{z}, \mathbf{p}) + \beta U_i(\mathbf{i}, \mathbf{z}, \mathbf{p}) + \gamma U_z(\mathbf{z}, \mathbf{l}) + \kappa U_l(\mathbf{l}, \mathbf{g})$$

es delicada, ya que ellos determinan la importancia específica que se le otorga a cada término de energía y por tanto, el resultado de la interacción local entre observaciones y etiquetas en cada paso del algoritmo. El comportamiento local acumulado marca a final de cuentas el desempeño general del algoritmo. Este no es un problema trivial, y en realidad, muchos autores lo han resuelto con estrategias más o menos heurísticas.

Ya se mencionó que el algoritmo de relajación se realiza en dos fases: en la primera, se ponderan (1) el *error modelo de movimiento-campo estimado* y (2) la propiedad de *continuidad del campo de etiquetas de segmentación*, mientras que en la segunda se ponderan (1) esta última característica, y (2) la *introducción de líneas de discontinuidad*. Hay una situación adicional: en la optimización se mezclan valores discretos y continuos. Tomando esto en consideración, el criterio utilizado para la selección de los parámetros fue el siguiente:

- Se escogieron los potenciales para los cliques del campo de líneas, ponderando las estructuras geométricas según se deseara favorecer o no su aparición en la segmentación. Otra opción para los potenciales de los cliques puede ser el uso de una estrategia similar a la descrita en [HEIT93].

- Una vez elegido este conjunto de valores, en  $U_1(\mathbf{l}, \mathbf{g})$  se ponderó el término  $l_{s_i}(1 - g_{s_i})$  por un factor de dos, para aumentar su efecto como evidencia parcial de presencia de discontinuidades de movimiento. Pueden usarse valores mayores [HEIT93], pero no se realizaron más pruebas a este respecto. Posteriormente, el valor de  $\kappa$  se hizo igual a uno.
- Un término clave es  $\gamma$ , porque pondera la energía de enlace entre las dos fases de optimización del problema. Debe, por tanto, contribuir adecuadamente en cada etapa de la relajación. Para la elección heurística de este parámetro, se realizaron algunas simulaciones del algoritmo, en donde sólo se consideraron las contribuciones de los términos  $U_z(\mathbf{z}, \mathbf{l})$  y  $U_1(\mathbf{l}, \mathbf{g})$ . Se observó que para valores de  $\gamma$  muy pequeños, el campo de líneas de discontinuidad no cambiaba. De esta forma, se eligió por prueba y error un valor de tres, que permite que el campo de líneas de discontinuidad se vaya actualizando.
- Finalmente se escogieron los valores de  $\alpha$  y  $\beta$ . En particular,  $\alpha$  debe ponderar un término continuo ( $U_d(\mathbf{d}_x, \mathbf{d}_y, \mathbf{z}, \mathbf{p})$ ) que puede variar mucho dependiendo de la secuencia particular a segmentar (magnitud y tipo de movimiento presente). Así, un valor relativamente elevado de  $\alpha$  exige que la estimación de movimiento se ajuste con mucha precisión al modelo de movimiento elegido, mientras que un valor muy pequeño produce una segmentación muy suave (continua). Esta situación se ha comentado en la literatura [MURR87]. Por otra parte, este coeficiente está relacionado con la variancia del término de error de observación  $\|\mathbf{d} - \mathbf{d}_0\|$ , y puede ser evaluado a partir del cálculo de dicha cantidad. Sin embargo, algunos autores no ven ventajas en el empleo de  $\sigma^2$ , y prefieren determinar  $\alpha$  de forma *ad hoc*, o bien mediante la definición de cocientes entre los diferentes términos de ponderación [KONR92]. En este trabajo, se utilizó un valor fijo de 10 tanto para  $\alpha$  como para  $\beta$ , tomando en cuenta que sólo uno de los dos términos,  $U_d(\mathbf{d}_x, \mathbf{d}_y, \mathbf{z}, \mathbf{p})$  y  $U_1(\mathbf{i}, \mathbf{z}, \mathbf{p})$ , contribuye a la vez a la energía total del sistema. La elección se efectuó a partir de algunas simulaciones con el algoritmo (no exhaustivas).

El criterio descrito se utilizó tomando como base a la secuencia sintética *Seqmill*. Para las demás secuencias de prueba se utilizaron los mismos parámetros. Como se apuntó, esto puede no ser óptimo, y por tanto se considera que el proceso de elección de parámetros puede mejorarse, a fin de incrementar el desempeño general del algoritmo.

## 7.2 Resultados y discusión.

Siguiendo el mismo orden de la primera parte, a continuación se presentan los resultados obtenidos con la secuencia sintética *Seqmill*, y con las secuencias reales *Miss América e Interview*.

### 7.2.1 Secuencia Seqmilt.

En primer lugar se presentan los resultados obtenidos con el algoritmo propuesto aplicado a la secuencia *Seqmilt*. La figura 7.1 muestra los datos de entrada (observaciones): (a) imagen original  $i$ , (b) imagen de bordes espaciales  $g$ , (c) segmentación inicial  $z_0$ , (d) imagen de pixels no compensados  $p$ , (e) campo de desplazamiento estimado  $d$ . Como se mencionó en el capítulo anterior, el campo de líneas se inicializa con el campo de gradientes espaciales, y la segmentación inicial se obtiene a partir de un detector de movimiento. Por otra parte, los campos estimados  $z$  y  $l$  aparecen en la figura 7.2. Adicionalmente, se muestra la superposición de la imagen original con las fronteras de la segmentación. Se usaron los parámetros de ponderación mencionados en la sección anterior ( $\alpha = 10$ ,  $\beta = 10$ ,  $\gamma = 3$  y  $\kappa = 1$ ). Se hará referencia a los objetos de la escena como R1 (rectángulo grande), R2 (rectángulo pequeño), C1 (cuadrado), y C2 (círculo)<sup>1</sup>. El análisis de las figuras 7.1 y 7.2 permite mencionar que:

- El campo de desplazamiento mostrado en su escala original contiene errores. Esto es evidente en las regiones de descubrimiento de R1, y en la zona de recubrimiento de R2. Pero existe otra fuente de error importante en la estimación, que es la inercia del movimiento (provocada por la naturaleza misma del método pel-recursivo). Este error, que se propaga en la dirección de barrido, es de mayor magnitud a la izquierda de R1, y también aparece en otras regiones circundantes a todos los objetos. Finalmente es importante hacer notar que en el interior de C1 y C2 el movimiento aparente estimado es nulo.
- La inicialización  $z_0$  es bastante buena, ya que de entrada se separan los objetos del fondo.
- En el mapa de pixels no compensados se alcanzan a distinguir puntos que corresponden a fronteras de movimiento (borde superior de R1 y R2, casi toda la frontera de C2), pero tales etiquetas no están presentes en todas las fronteras. En particular, se observa que en las regiones en las que el campo está mal estimado no predominan las etiquetas  $p_s = 1$ . Ya antes se había dicho que el campo  $p$  representaba sólo una medida de confianza parcial en cuanto a la consistencia del campo con la realidad; aquí se ven con claridad sus limitaciones.
- La superposición de la trama original y la segmentación pone en evidencia la precisión del método para ajustarse a las discontinuidades de movimiento. Este importante resultado no se reporta en otros trabajos de segmentación. En nuestro caso, la segmentación final obtenida muestra que:

<sup>1</sup>Por cuestión de claridad, las referencias de lateralidad de los objetos en este trabajo siempre se consideran desde el punto de vista del *objeto*.

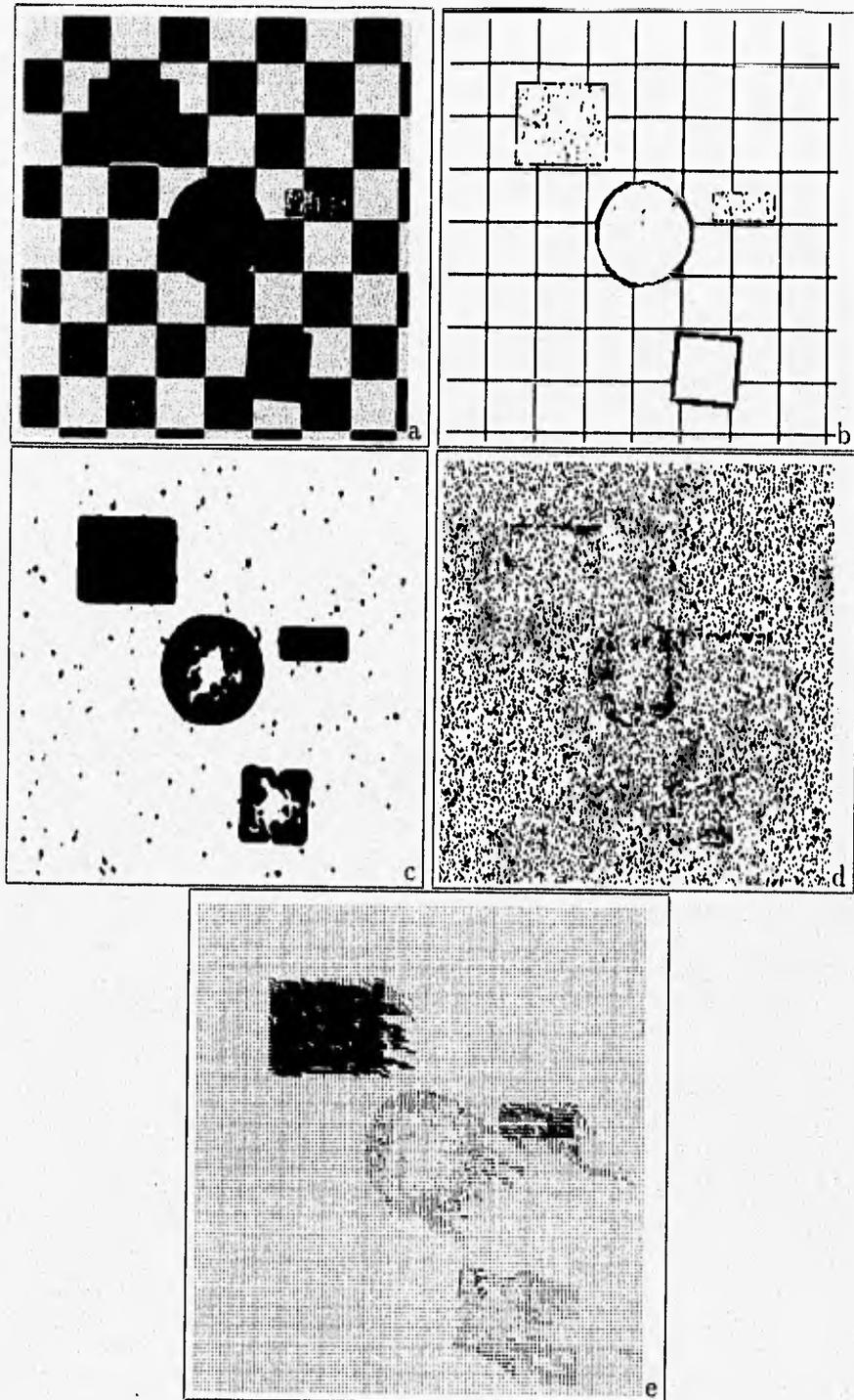


Figura 7.1: Información de entrada al algoritmo de segmentación basada en movimiento. Secuencia *Seqmilt*. Tramas 1 y 2. (a) Campo **i**, (b) campo **g**, (c) inicialización **z<sub>0</sub>**, (d) campo **p**, (e) campo **d**. El campo de desplazamiento aparece submuestreado cada 2 pixels y en su tamaño original.

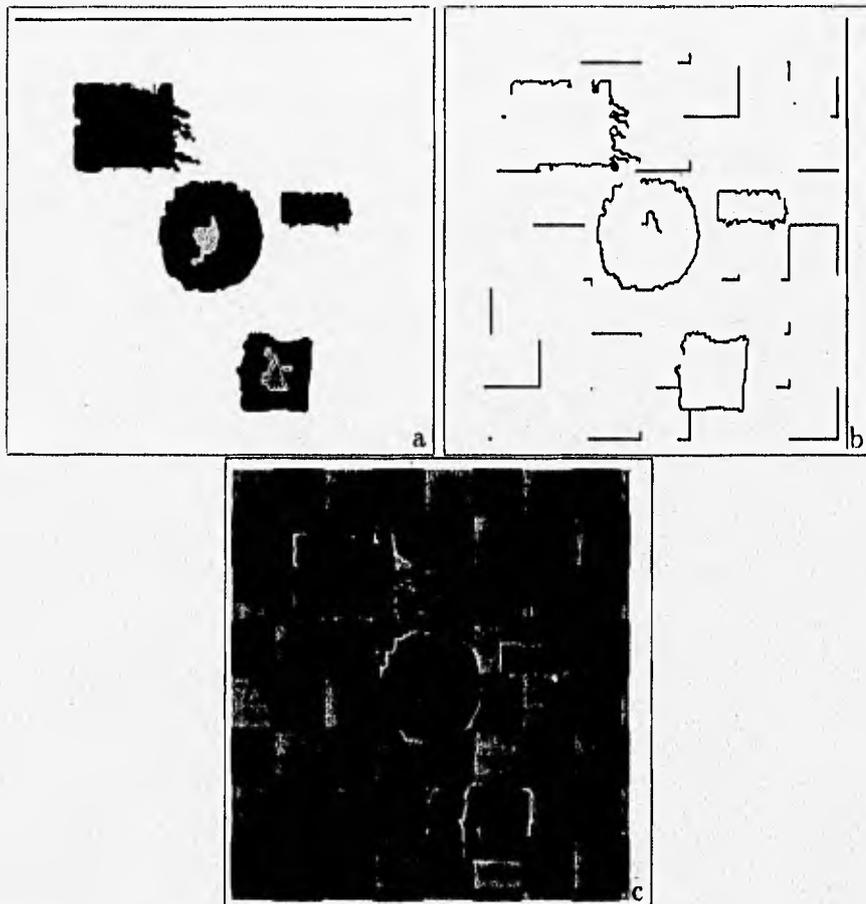


Figura 7.2: Información producida por el algoritmo de segmentación basada en movimiento. Secuencia *Seqmilt*. 5 iteraciones. (a) Campo de etiquetas de segmentación  $z$ ; número final de regiones: 7, (b) campo de líneas de discontinuidad de movimiento  $l$ , (c) superposición de la trama 2 de la secuencia y la imagen de fronteras de la segmentación  $z$ . El contraste de la imagen original se ha cambiado para visualizar mejor las fronteras entre las regiones.

- Las fronteras de la segmentación se ajustan bastante bien en R2 y C2.
- El ajuste es menos preciso en C1.
- Las regiones al interior de C1 y C2 se mantienen.
- En R1 el ajuste es bueno en la frontera inferior, y en parte de la superior, pero falla en las demás fronteras. Por una parte, las regiones de descubrimiento a la derecha de R1 se están fusionando con el objeto en movimiento; este hecho no ocurre, en cambio, en su frontera superior. Por otro lado, dentro de la región correspondiente a R1 aparecen subregiones que no pertenecen al objeto, y al mismo tiempo, una pequeña subregión del mismo queda fuera de la región. Estas situaciones no corresponden con lo deseado.
- ¿Qué ocurre? Lo que se está observando es el efecto combinado de (a) la existencia de ruido en las observaciones (el desplazamiento fue mal estimado en las regiones problemáticas), y (b) la ausencia de una medida de confianza que permita evitar el uso de esta información errónea y emplear en cambio otra información más confiable (como la intensidad de la imagen). La corrupción del campo se está viendo reflejado en la segmentación.

- Con respecto al campo de líneas de discontinuidad obtenido, se observa que se marcan más o menos bien las fronteras de movimiento de los objetos, con los errores ya mencionados. Adicionalmente, persisten algunas líneas de gradiente espacial que no corresponden al movimiento. Sin embargo, estas líneas erróneas tienden a disminuir si el número de iteraciones aumenta.

Hay que mencionar que el algoritmo converge después de 5 iteraciones aproximadamente. Lo mismo ocurre con las otras secuencias de prueba empleadas. Si bien el número de iteraciones es pequeño, el tiempo de cálculo es considerable, y parece muy grande para un enfoque de relajación determinístico. Este hecho se debe fundamentalmente a dos factores, uno propio al algoritmo (el término  $U_d(\mathbf{d}_x, \mathbf{d}_y, \mathbf{z}, \mathbf{p})$  es continuo, y todos los cálculos se realizan en punto flotante), y otro asociado a la implantación particular del método (en la programación no se realizó ningún esfuerzo para hacer más veloz el algoritmo; de hecho, algunas rutinas programadas resultan bastante lentas *per se*). Después de 4 ó 5 iteraciones se observa que el campo  $\mathbf{z}$  prácticamente no sufre modificaciones, pero el campo  $\mathbf{l}$  sí lo hace durante algunas iteraciones más, y tiende a concentrarse en las fronteras de las regiones de la segmentación  $\mathbf{z}$  ya estable.

### **Cambios en la inicialización del campo de líneas de discontinuidad.**

Se mencionó en el capítulo anterior que otra opción para inicializar el campo de líneas  $\mathbf{l}_0$  era utilizar las fronteras de la segmentación inicial  $\mathbf{z}_0$ . En la figura 7.3 se muestra el campo final de líneas obtenido con esta nueva inicialización. Como se ve,

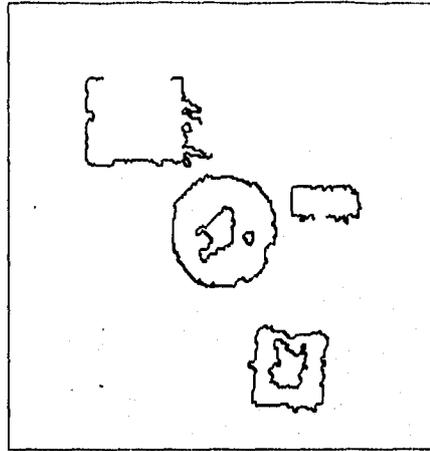


Figura 7.3: Campo final de líneas de discontinuidad obtenido con  $l_0$  igual a las fronteras de la segmentación inicial  $z_0$ .

$l$  sólo mantiene las líneas correspondientes a las regiones en movimiento. Por otra parte, la segmentación final  $z$  es similar a la obtenida a partir de la inicialización  $l_0 = g$ .

#### Ajuste de los coeficientes de ponderación.

Se mencionó en la sección 7.1 que la elección de los coeficientes de peso del modelo markoviano es muy importante. En particular, el parámetro  $\alpha$  pondera la importancia que se le da a la precisión entre las observaciones y el modelo de movimiento. Para ilustrar este hecho, se muestra en la figura 7.4 el resultado de ajustar el parámetro  $\alpha$  a un valor de 1, para producir una segmentación más suave. Como se ve, las fronteras de la segmentación se ajustan mejor ahora para R1, aunque se observa que en R2 Y C1 el ajuste es inferior que en el caso  $\alpha = 10$ . Este resultado indica que para que se logre un ajuste de fronteras de movimiento muy preciso, debe contarse con dos cosas: (a) un campo de desplazamiento muy bueno y (b) un mecanismo que permita disminuir la influencia de los errores en la estimación. En este trabajo no se realizaron más pruebas sobre la optimización de los parámetros de ponderación.

#### Uso de la información de niveles de gris.

Para ilustrar las ventajas que conlleva combinar el campo de pixels no compensados con la información de nivel de gris, se presenta la figura 7.5. En este caso, se modificaron los parámetros de la secuencia *Seqmilt* para generar un movimiento mucho mayor. Considérese la región superior del rectángulo móvil. Como se ve en el campo de desplazamiento estimado, el algoritmo pel-recursivo produce errores en las regiones de descubrimiento que se propagan a otras regiones. Ahora bien, en

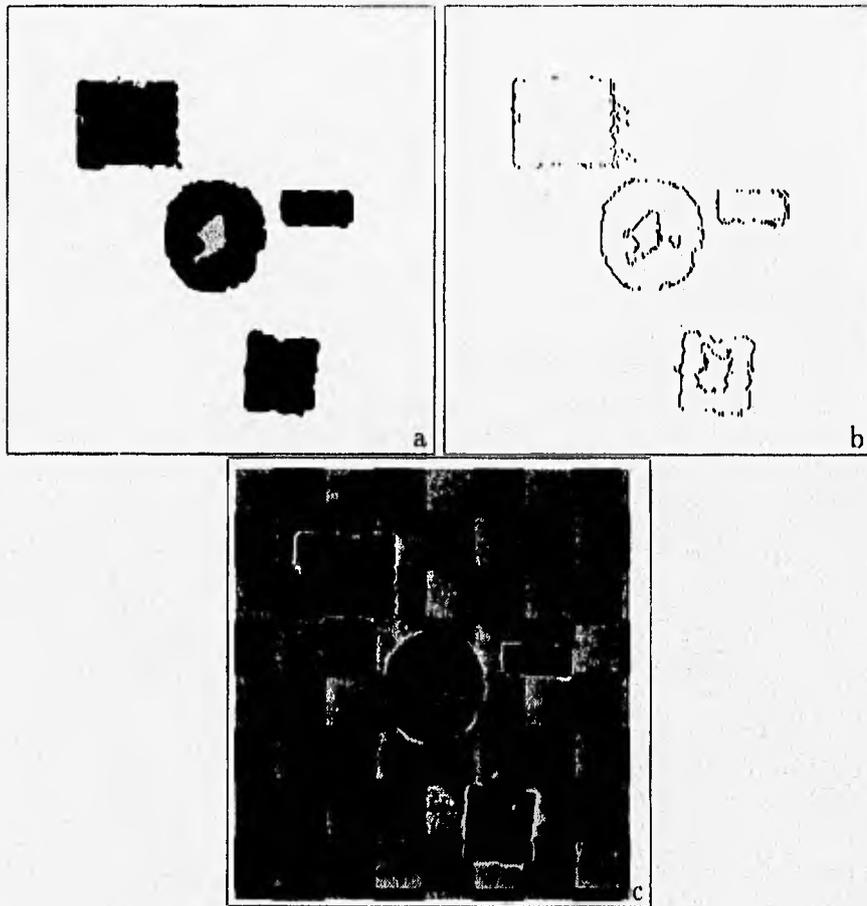


Figura 7.4: Segmentación basada en movimiento. Cambio en el parámetro de ponderación  $\alpha$ . (a) Campo de etiquetas de segmentación  $z$ , (b) campo de líneas de discontinuidad de movimiento  $l$ , (c) superposición de la trama 2 de la secuencia y la imagen de fronteras de la segmentación  $z$ .

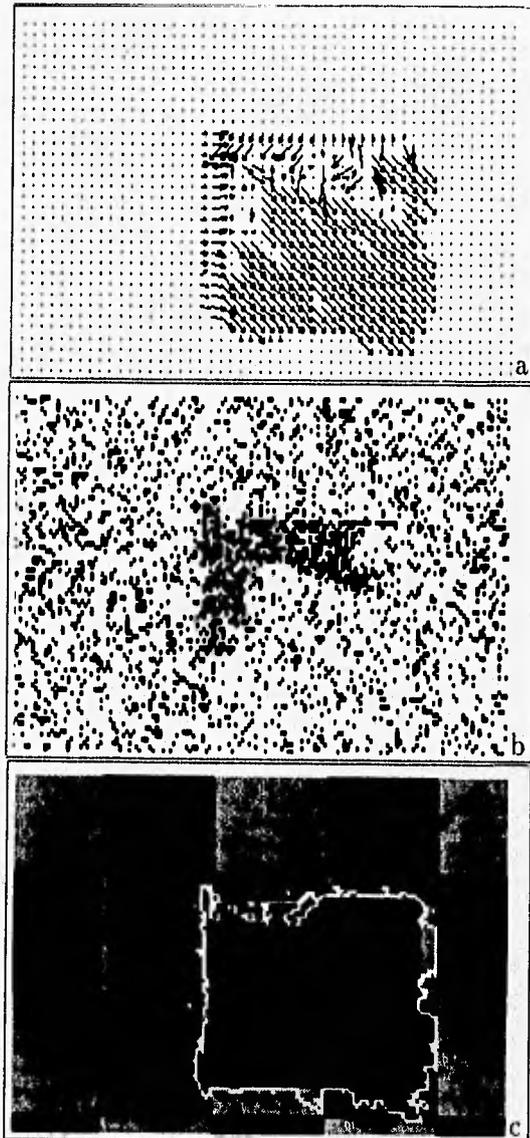


Figura 7.5: Segmentación basada en movimiento. Uso de la información de niveles de gris. (a) Campo de movimiento estimado  $\mathbf{d}$ , (b) campo de pixels no compensados  $\mathbf{p}$ , (c) superposición de la trama original y la segmentación  $\mathbf{z}$ .

este caso la imagen de pixels no compensados *sí* proporciona información útil, ya que justamente en la región en la que el método BD falla, queda constancia del error mediante la etiqueta  $p_s = 1$ . Esto ocasiona que el algoritmo de segmentación tome en cuenta a la intensidad de la imagen (campo  $i$ ), y promueva la continuidad de la segmentación en base al nivel de gris, evitando la introducción de errores debidos al campo de movimiento. El resultado no es perfecto, pero sí muestra claramente el beneficio de utilizar información más confiable cuando las observaciones principales son erróneas.

### 7.2.2 Secuencia Miss América.

Para la segunda secuencia de prueba, se presentan en la figura 7.6 las observaciones, y en la figura 7.7 las etiquetas. En este caso, se observa lo siguiente:

- El campo  $d$  es predominantemente translacional. Por la inexistencia de gradientes fuertes en la secuencia (a excepción del rostro y la silueta del torso de la mujer), en regiones como la cabeza no se estima movimiento. Por otra parte, si se observa este campo en su escala original, se aprecia nuevamente el efecto de inercia de movimiento en la dirección de barrido, sobre todo en el lado izquierdo del rostro de la mujer.
- El detector de movimiento que se usa como inicialización para la segmentación resulta adecuado, sobre todo si se compara con las características del campo estimado  $d$ .
- El campo  $p$  aparece distribuido sobre todo el sujeto; se alcanza a distinguir la silueta de la mujer (cabello, hombros, cuello) pero no se marca alguna estructura específica.
- Aquí, a diferencia de la secuencia anterior, resulta más difícil hablar de una *segmentación ideal*: es obvio que deben separarse las regiones con movimiento muy pequeño o nulo de las regiones con movimiento grande, pero no se puede decir lo mismo en el caso de regiones adyacentes con movimiento ligeramente diferente.
- La segmentación obtenida se considera buena. Sin embargo, de la imagen superpuesta, se aprecia que las regiones de descubrimiento están siendo absorbidas por la región en movimiento (cuello y cabeza) y por tanto, las fronteras de la segmentación no se ajustan con exactitud a las fronteras reales; asimismo, a la izquierda del rostro de la mujer el ajuste tampoco es preciso. Existen otras regiones donde el ajuste es mejor.
- El campo de líneas de discontinuidad prácticamente eliminó todos los gradientes verticales y horizontales y dejó solamente algunas líneas aisladas.

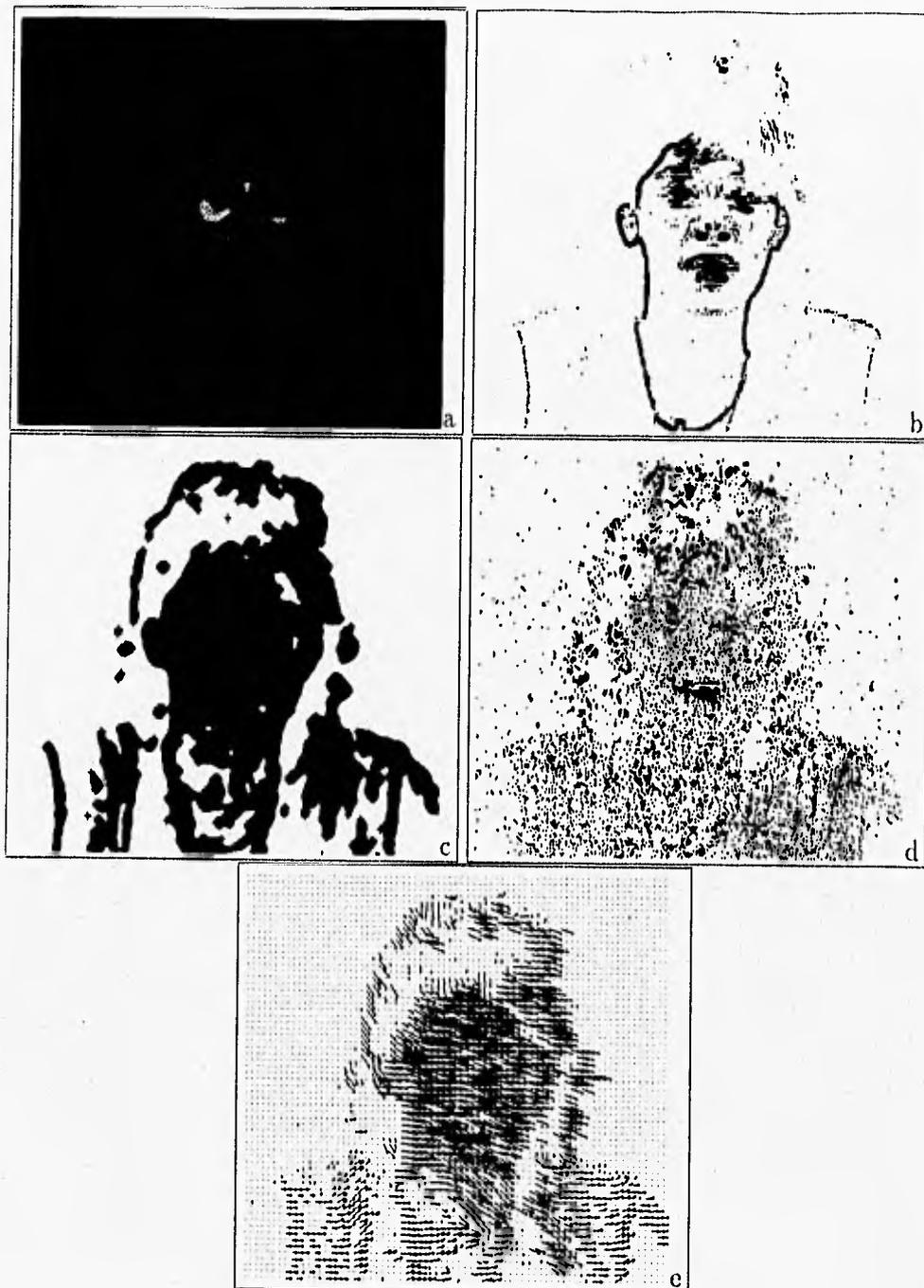


Figura 7.6: Información de entrada al algoritmo de segmentación basada en movimiento. Secuencia *Miss América*. Tramas 38 y 39. (a) Campo  $i$ , (b) campo  $g$ , (c) inicialización  $z_0$  (detector de movimiento), (d) campo  $p$ , (e) campo  $d$ . El campo inicial  $l_0$  se eligió igual a  $g$ . El campo de desplazamiento aparece submuestreado cada 4 pixels, y al doble de su escala original.



Figura 7.7: Información producida por el algoritmo de segmentación basada en movimiento. Secuencia *Miss América*. (a) Campo de etiquetas de segmentación  $z$ ; número final de regiones: 36, (b) campo de líneas de discontinuidad de movimiento  $l$ , (c) superposición de la trama 39 de la secuencia y la imagen de fronteras de la segmentación  $z$ .

- Los resultados hacen pensar que, si bien la segmentación final es aceptable, probablemente se pueda mejorar con un procesamiento adecuado de las regiones de oclusión (a través del manejo de las etiquetas provisionales), o con una elección mejor del conjunto de coeficientes de ponderación.

### 7.2.3 Secuencia Interview.

En la figura 7.8 se muestran las observaciones introducidas al modelo markoviano. Los campos iniciales  $l_0$  y  $z_0$  y los parámetros de ponderación se eligen como ya se mencionó antes. Los resultados obtenidos se muestran en la figura 7.9. Para propósitos de discusión, la mujer que aparece a la izquierda en la imagen se denominará mujer A, y la que aparece a la derecha, mujer B. Con respecto a los resultados, puede comentarse que:

- La elección de la segmentación inicial fue suficientemente buena. Como se observa, las regiones móviles se detectan justo donde existen diferencias temporales fuertes entre una trama y otra, como en la mujer B, en las fronteras entre los objetos y el fondo de la escena, y en el diseño a rayas en la ropa de la mujer A.
- El objeto de mayor movimiento (mujer B) fue segmentado correctamente, aunque existen regiones interiores al mismo.
- En general, todas las demás regiones iniciales fueron reducidas, pero no eliminadas.
- Se aprecia que el campo de líneas de discontinuidad de movimiento obtenido tiende a concentrarse en las fronteras de las regiones de la segmentación. Para esta secuencia es clara su utilidad como un producto secundario del proceso de segmentación.
- En general, la segmentación se ajusta bien a las fronteras de movimiento. Haciendo referencia particular a la mujer B, el algoritmo no ajusta bien las fronteras de la segmentación en la región derecha de su cabeza. También se aprecia una tendencia a ampliar las regiones de movimiento en la dirección del barrido (véase la región izquierda de la cabeza). En cambio, el algoritmo se ajusta bien, por ejemplo, en el brazo derecho y el tronco.
- ¿A qué se debe que falle el algoritmo? Nuevamente la explicación está en los errores en la estimación del campo de desplazamiento. Para el caso de la región derecha de la cabeza de la mujer B, obsérvese la región correspondiente en el campo de movimiento. Es claro que en esa región el campo fue mal estimado, no tanto en el sentido de calidad de reconstrucción (porque en esa región las etiquetas de "no compensado" no se concentran), sino de coherencia con el campo "verdadero". Ahora bien, la tendencia a ampliar las regiones se debe esencialmente al efecto de inercia de movimiento ya discutido.

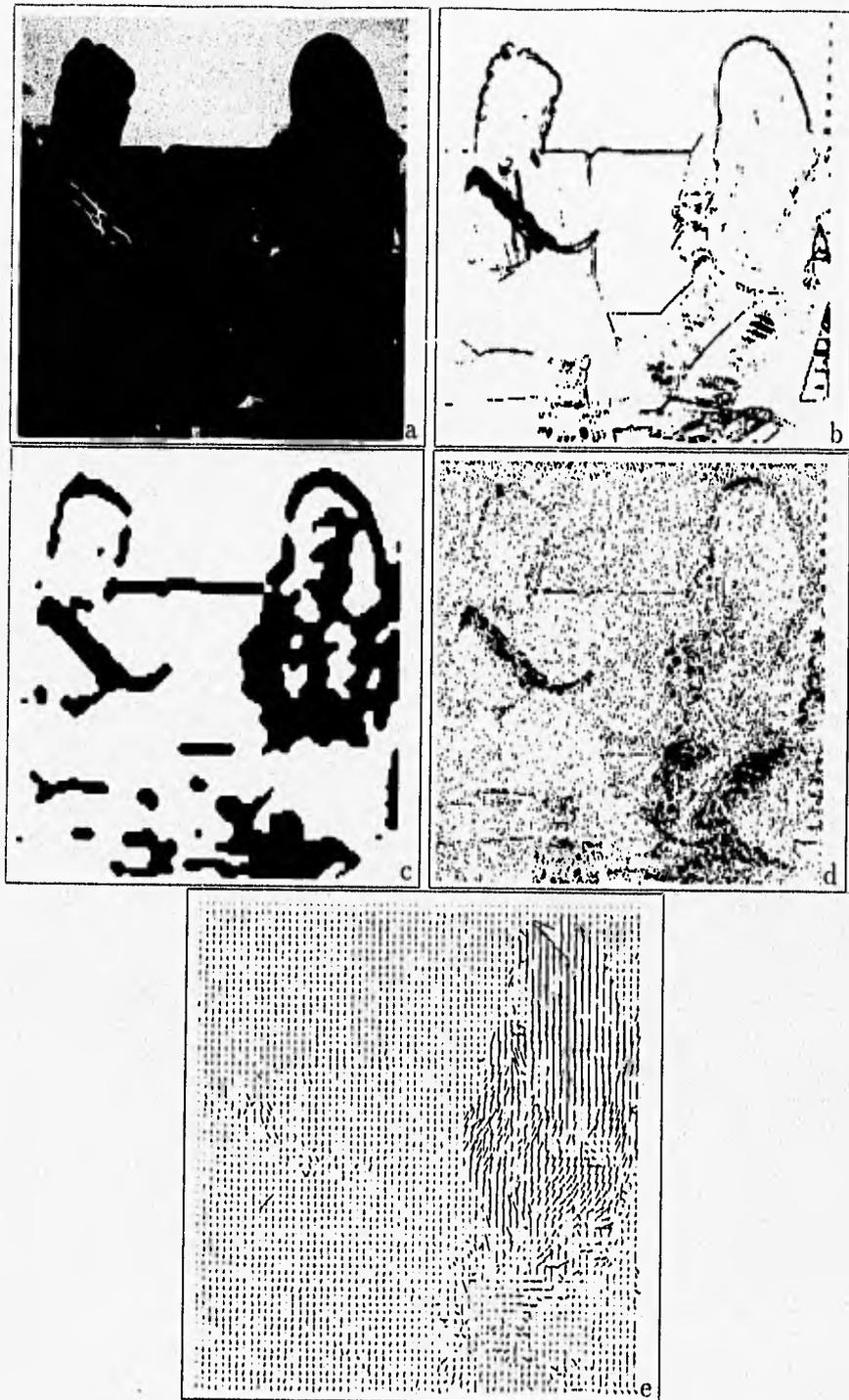


Figura 7.8: Información de entrada al algoritmo de segmentación basada en movimiento. Secuencia *Interview*. Tramas 6 y 7. (a) Campo  $i$ , (b) campo  $g$ , (c) inicialización  $z_0$ , (d) campo  $p$ , (e) campo  $d$ . El campo inicial  $l_0$  se eligió igual a  $g$ . El campo de desplazamiento aparece submuestreado cada 4 pixels, y amplificado por 3.

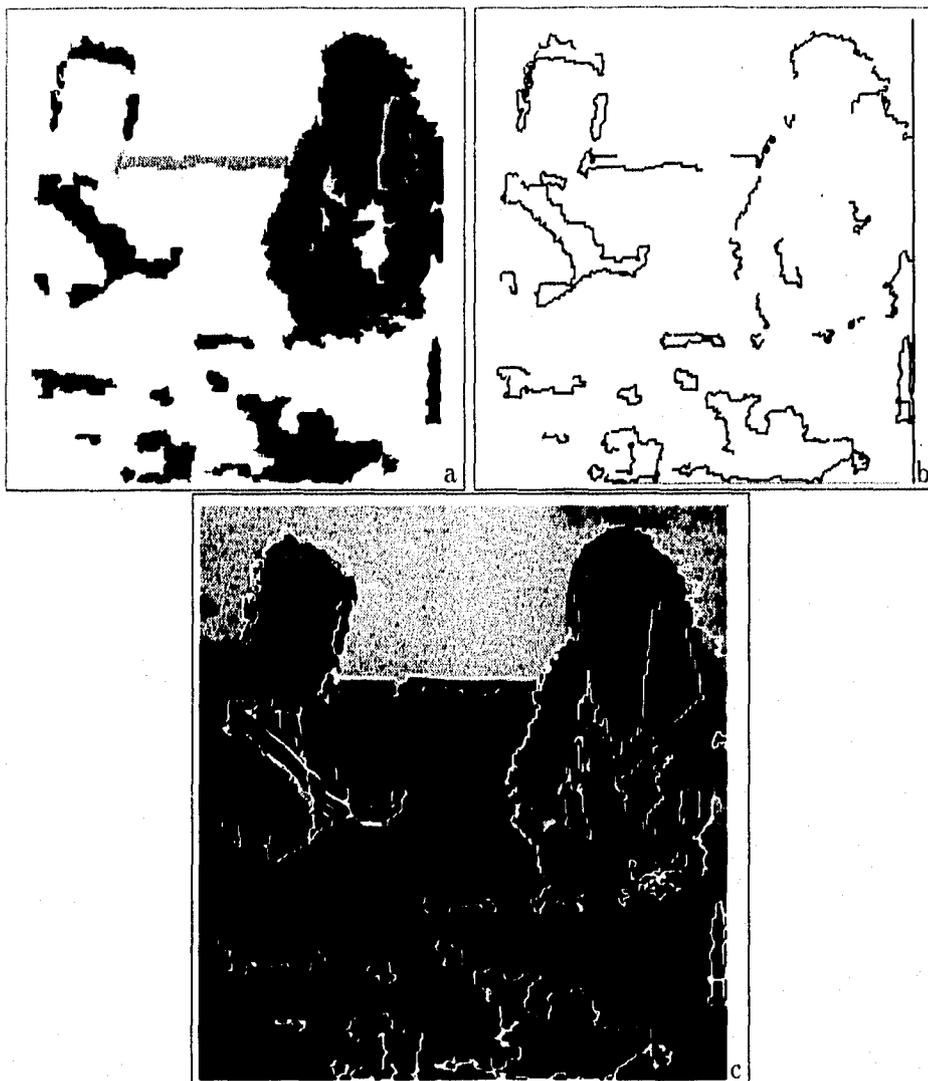


Figura 7.9: Información producida por el algoritmo de segmentación basada en movimiento. Secuencia *Interview*. (a) Campo de etiquetas de segmentación  $z$ ; número final de regiones: 28, (b) campo de líneas de discontinuidad de movimiento  $l$ , (c) superposición de la trama 7 de la secuencia y la imagen de fronteras de la segmentación  $z$ . El contraste de la imagen original se ha cambiado para visualizar mejor las fronteras entre las regiones.

- Se concluye que los resultados son satisfactorios.

#### **7.2.4 Recapitulación.**

Hasta aquí se presentaron los resultados obtenidos con el algoritmo propuesto de SBM. Se encontró que:

- En general, las segmentaciones obtenidas son buenas y concuerdan con la realidad.
- La introducción de la información de nivel de gris en los casos en los que falla la estimación es benéfica, siempre y cuando existan indicadores de confianza adecuados.
- En las zonas donde la estimación de desplazamiento es mala en cuanto a consistencia con la realidad, aunque buena en cuanto a calidad de reconstrucción, se presentan errores en la segmentación. Recuérdese la relación estimación-segmentación de movimiento.
- La elección de los coeficientes de ponderación del modelo markoviano es delicada.
- Se considera que el método puede mejorarse, dentro del mismo esquema propuesto.

En la sección siguiente se sugieren algunas ideas para mejorar los resultados obtenidos, y se discuten un par de cuestiones muy importantes relacionadas con la continuación del presente trabajo.

### **7.3 Perspectivas del trabajo.**

#### **7.3.1 Mejoras al algoritmo.**

Como ya se mencionó, el algoritmo de segmentación basada en el campo de movimiento calculado por el método pel-recursivo basado en estimación de Wiener puede mejorarse de varias maneras. A continuación se enlistan algunas ideas que pueden aplicarse y que probablemente abran otras líneas de trabajo:

- Con respecto a la estimación de movimiento.
  - Incluir en la lógica del algoritmo pel-recursivo algún criterio para disminuir el efecto de inercia de movimiento.

- Se ha demostrado que es posible determinar cuantitativamente la pérdida de precisión en la estimación en áreas ambiguas [KEAR87]. Una posibilidad es entonces encontrar buenas medidas de confianza con respecto a la calidad de la estimación del desplazamiento (físicamente consistente), y no solamente en el sentido de reconstrucción de la imagen. Dichas medidas pueden usarse para ponderar la contribución de las estimaciones de movimiento, o bien para filtrar aquellas regiones en donde se presentan estimaciones pobres del campo. Este concepto se ha utilizado recientemente [HEIT93], [BARR94], [ZHEN95].
  - Utilizar un esquema de predicción combinado espacial espacio-temporal para favorecer la continuidad del campo de desplazamiento a lo largo de la secuencia [MOOR87].
  - En este mismo sentido, algunos autores recomiendan utilizar filtrado espacio-temporal como preprocesamiento para mejorar la estimación de los campos [BARR94].
- Con respecto a la segmentación.
    - Optimizar los parámetros de ponderación del modelo markoviano. Sería muy conveniente hallar una forma más inteligente de elegir el conjunto de parámetros para cada secuencia, o incluso lograr que éstos fueran adaptativos dentro de una misma secuencia.
    - Dentro del modelo, utilizar no solamente sistemas de vecindad espacial para el campo de sitios, sino espacio-temporal [MURR87] o espacio-temporal con información de movimiento [CHAN93].
    - Incluir en el modelo las nuevas medidas de confianza.
    - Encontrar un algoritmo más eficiente para el tratamiento de las etiquetas provisionales, a fin de que contribuyan en mayor medida en el procesamiento de las regiones de oclusión.
    - Reducir el tiempo de cálculo. Hay que decir, sin embargo, que la sofisticación del modelo conlleva un aumento en la carga computacional, por lo que el compromiso complejidad del modelo-tiempo de cálculo es inevitable.
  - Con respecto al binomio estimación-segmentación.
    - Desarrollar una estrategia iterativa estimación-segmentación, de tipo multi-resolución. Este tipo de enfoques ya ha demostrado su utilidad dentro del análisis de movimiento [BAAZ91b], [ANAN93], [KONR92], [HEIT93], [TEKA95], por las distintas ventajas que ofrece: mejor manejo de desplazamientos grandes, buena inicialización de un nivel a otro de la estructura piramidal, y reducción del tiempo de cálculo.

### 7.3.2 Criterios de fusión de regiones.

En el método de segmentación propuesto no se incluyó ningún algoritmo de *fusión de regiones*, aunque los criterios que la realizan se han utilizado ampliamente [ADIV85], [FRAN91], [NICO92b], [WANG93], [GARC95a]; se ha preferido en este trabajo que la fusión de datos se realice solamente a nivel de pixels, dejando evolucionar libremente al algoritmo markoviano.

En la secuencia *Interview*, al observar los parámetros de movimiento  $\Theta_{MLS}$  de las regiones obtenidas, se encuentra que, por ejemplo, hay regiones que se podrían unir con el fondo de la escena bajo un criterio del tipo *inspección de hipótesis* (*hypothesis testing*), ya que los parámetros de estas regiones son similares. Sin embargo, la elección de un método de fusión/división de regiones (después de cada iteración del algoritmo, o al final de todas las iteraciones) es un problema delicado, que conviene discutir a continuación.

Generalmente, la elección de un *criterio de decisión* de fusión de regiones se basa en dos tendencias principales: *máxima similitud entre los parámetros del modelo de movimiento*, y *mínimo error de reconstrucción*. Este criterio, representado por una cantidad  $E_F$ , permite entonces unir dos regiones  $R_{\Theta_1}$  y  $R_{\Theta_2}$  en una sola  $R_{\Theta_3}$ , o mantenerlas separadas. Estas tendencias se han utilizado para aplicaciones de interpretación [FRAN91], y codificación [NICO92b], [GARC95a].

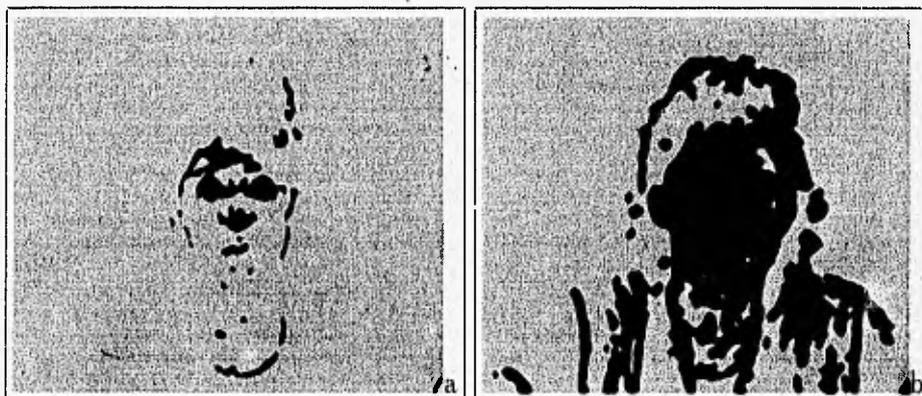


Figura 7.10: Aspectos importantes para la elección de un criterio de fusión de regiones. Secuencia *Miss América*. (a) Hay tramas en las que el movimiento local predomina. (b) En otras tramas, el movimiento es global. Todo el rostro es homogéneo en base al movimiento. ¿Qué conviene más: mantener siempre separadas aquellas regiones que a veces son homogéneas en cuanto al movimiento y a veces no, o fusionarlas y crearlas cada vez que sea necesario?

La definición de un criterio de decisión debe tomar en cuenta otros aspectos. Considérese la secuencia *Miss América*. Como ya se mostró, existen partes de la

secuencia en las cuales el movimiento es muy localizado (ojos, boca, nariz), y hay otras tramas en donde el movimiento de la mujer es de tipo global. Supóngase que para la SBM se parte de una segmentación inicial  $z_0$  de tipo espacial; en ella, las diferentes partes del rostro aparecen como regiones espacialmente ajenas. Ahora bien, para el caso de *movimiento global*, se observó ya que todo el rostro corresponde aproximadamente a una sola región (figura 7.10(b)); por tanto, las pequeñas regiones correspondientes a los ojos, boca y nariz deberían fusionarse con el resto del objeto. Pero, ¿qué ocurre cuando la secuencia evoluciona y aparece nuevamente el caso de *movimiento local*? Según lo anterior, la región del rostro debería dividirse para permitir la separación de las regiones correspondientes a los ojos y la boca, ya que ahora tienen un movimiento *distinto*. Este ejemplo, común en secuencias reales, plantea un problema interesante: ¿Qué conviene más: un algoritmo que continuamente esté creando y haciendo desaparecer regiones (fusión/división) entre cada par de tramas de la secuencia, o un algoritmo que mantenga más o menos el mismo número de regiones, que las vaya actualizando (deformando) conforme la secuencia evoluciona, y que permita el surgimiento de nuevas regiones únicamente cuando sea preciso (como en el caso de regiones de descubrimiento)? La respuesta a esta pregunta otra vez depende de varios factores, y el problema no puede resolverse de manera única sin pensar antes en la aplicación que quiera dársele al algoritmo de segmentación. Así, por ejemplo, para propósitos de seguimiento de objetos o análisis cualitativo de escena, puede ser suficiente con obtener una segmentación "gruesa" (pocas regiones en la segmentación, y un número pequeño de regiones para cada objeto de la escena), mientras que para codificación resulta ventajoso mantener un número mayor de regiones.

En particular, para codificación orientada a regiones, debe tomarse en cuenta la eficiencia de la compresión de la información (estimación/segmentación) para un nivel de degradación dado, y viceversa. Este aspecto debe incluir, por tanto, la *optimización de la segmentación* para establecer un buen compromiso *compresión-reconstrucción*, que constituye por sí misma toda una línea de investigación. Un principio que se ha usado recientemente como herramienta para análisis de imágenes y para compresión de secuencias es el de MDL (*minimum description length*) [ZHEN95]. El MDL minimiza el costo (longitud) de la codificación de los datos, mediante una función donde se involucran tanto el número de bits necesarios para codificar el error de predicción (calidad de reconstrucción de la secuencia), como el costo de la codificación de la segmentación (fronteras entre regiones). La descripción y análisis de este principio está más allá de los propósitos de este trabajo, pero se comenta para ilustrar que la elección de un criterio óptimo de fusión de regiones constituye un problema abierto.

### 7.3.3 Posibles aplicaciones.

Se mencionó en la primera parte de la tesis que una posible aplicación de la segmentación del campo de vectores de correspondencia es la compresión de secuen-

cias de imágenes *basada en regiones* [GARC95a], [TEKA95], [ZHEN95]. En este esquema de compresión, en primer lugar se extraen las regiones (se determinan los parámetros que describen el movimiento de cada una de ellas). Posteriormente, los parámetros se codifican y transmiten junto con imágenes de error de predicción con compensación de movimiento. También debe codificarse y transmitirse la información referente a los contornos de la segmentación. En el receptor, se sintetiza el campo de movimiento a partir de la información recibida, y luego este campo sintetizado se usa para reconstruir cada trama de la secuencia.

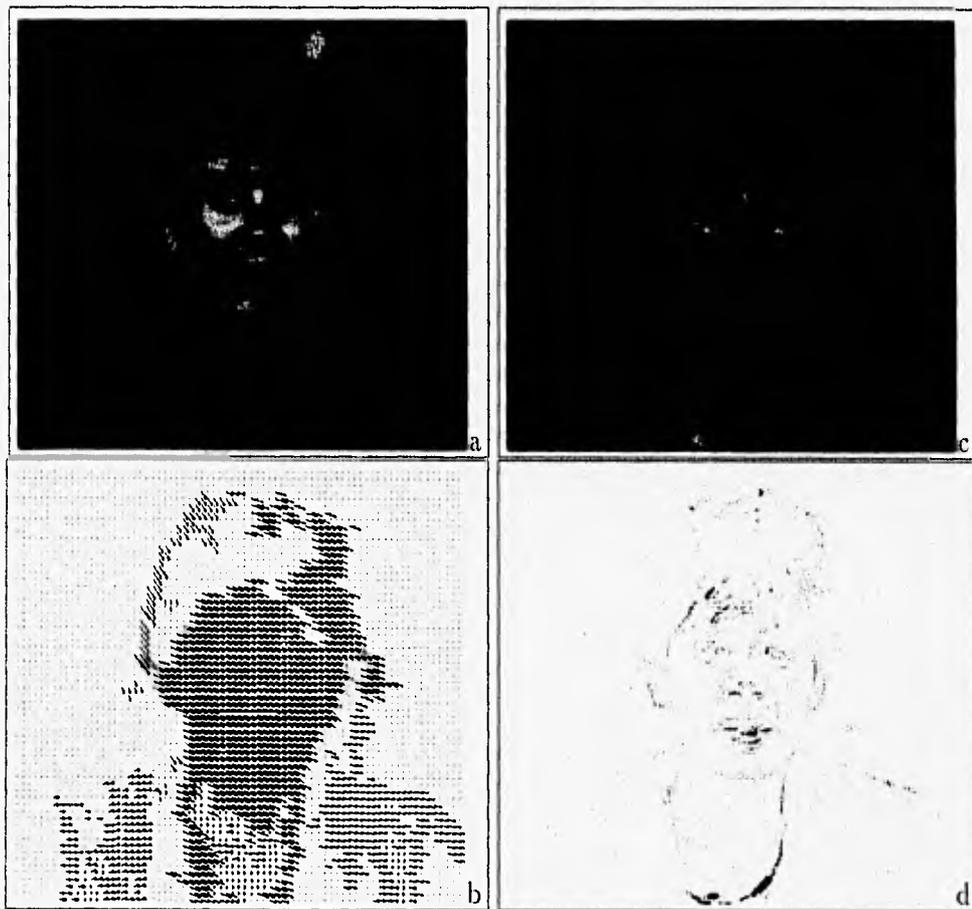


Figura 7.11: Reconstrucción de la secuencia *Miss América* a partir de la SBM. (a) Trama 39 original. (b) Campo sintetizado a partir de la segmentación correspondiente a las tramas 38 y 39 (submuestreado cada 4 pixels, amplificado por 2). (c) Trama 39 reconstruida. (d) Valor absoluto del error de reconstrucción. PSNR = 34.9 dB. El error se muestra en negativo para su visualización.

El problema de la codificación de secuencias utilizando información de movimiento puede verse como un problema de optimización de un criterio costo-beneficio,

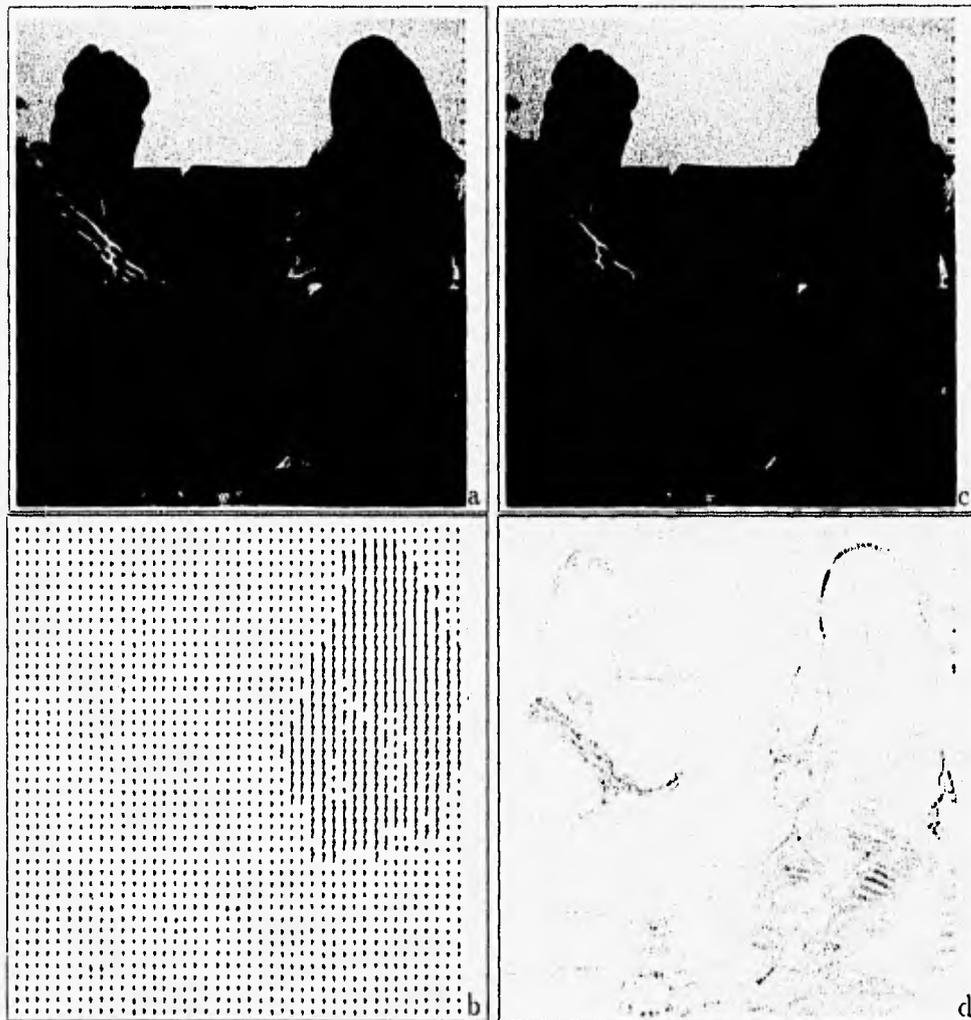


Figura 7.12: Reconstrucción de la secuencia *Interview* a partir de la SBM. (a) Trama 7 original. (b) Campo sintetizado a partir de la segmentación correspondiente a las tramas 6 y 7 (submuestreado cada 4 pixels, amplificado por 2). (c) Trama 7 reconstruida. (d) Valor absoluto del error de reconstrucción. PSNR = 32.8 dB. El error se muestra en negativo para su visualización.

en el que se intercambia calidad de reconstrucción por una mayor reducción de los datos a transmitir. La clave de codificar por regiones consiste en representar cada región homogénea en cuanto al movimiento - alta correlación entre sus vectores de desplazamiento - por un solo vector de parámetros, con lo que se pueden obtener tasas de compresión más altas que las obtenidas por algoritmos clásicos. Además, se pueden asignar diferentes tasas de codificación a cada región dependiendo de su importancia (contribución para la calidad visual de las imágenes reconstruidas).

Para ilustrar una aplicación potencial de este tipo, se muestran en las figuras 7.11 y 7.12 algunos resultados obtenidos para las secuencias *Miss América e Interview*. Las imágenes se reconstruyen (compensan) a partir de los campos sintetizados por la segmentación. Como se ve, las imágenes reconstruidas tienen muy buena calidad visual, con PSNR de 34.9 dB y 32.8 dB respectivamente. Una ventaja de este enfoque de codificación es que, como las fronteras de los segmentos regularmente coinciden con bordes espaciales de la imagen, se reducen considerablemente los artefactos que se suelen apreciar con métodos de compresión tradicionales como los de puesta en correspondencia entre bloques [TEKA95].

Como se mencionó en la subsección anterior, el estudio de un esquema particular para codificar conjuntamente la información de movimiento y de segmentación está fuera de los objetivos de este trabajo, pero marca otra línea de desarrollo futuro.

## 7.4 Conclusiones.

En este capítulo se presentaron los resultados obtenidos con el algoritmo propuesto de segmentación basada en movimiento, experimentando con distintos tipos de secuencias de imágenes, a fin de realizar una evaluación cualitativa del desempeño del método bajo diferentes condiciones. En especial, se evaluó cualitativamente la precisión con la que la segmentación se ajusta a las fronteras reales de movimiento.

La aplicación de la teoría de campos aleatorios de Markov al problema de la segmentación de movimiento, que permitió la inclusión de distintas fuentes de información y el modelado de sus interacciones mediante restricciones contextuales, produjo resultados de buena calidad. Sin embargo, se corroboró en la práctica que la condición principal para lograr una segmentación precisa es contar con un buen campo de movimiento, y con información confiable acerca de la calidad del mismo. Por tanto, se piensa que los resultados pueden ser mejorados dentro del mismo marco general analizado, y en base a ello se mencionaron algunas ideas que pueden marcar la continuidad de esta línea de trabajo y determinar aplicaciones futuras.

## Capítulo 8

### Conclusiones generales.

Este trabajo se dedicó al análisis, estudio e implantación de dos de los principales objetivos del análisis de movimiento en secuencias de imágenes: la estimación y la segmentación de movimiento aparente. Ambos problemas son etapas clave tanto para labores de visión computacional como para procesamiento de secuencias y, al igual que muchos otros interesantes dominios de estas dos ramas de la ingeniería, pueden concebirse dentro de un enfoque general de problemas de optimización. En nuestro caso, el enfoque utilizado fue el siguiente:

- Generar, a partir de dos tramas de una secuencia, un campo denso de desplazamiento aparente.
- Emplear el campo generado como base para efectuar la segmentación.

La estimación del campo de vectores de correspondencia se realizó con métodos pel-recursivos clásicos. Se estudiaron e implantaron cuatro de ellos, y posteriormente se realizó un análisis cuantitativo y cualitativo de su desempeño. Todos los algoritmos demostraron ser excelentes para la reconstrucción (por compensación de movimiento) de secuencias variadas y complejas, ya que se basan en la minimización del error de reconstrucción, pero sólo uno de ellos - basado en teoría de estimación de Wiener - produjo campos con significado físico valioso, y por tanto se eligió como la fuente de información básica para la implantación del segundo proceso.

Para la segmentación basada en movimiento, se optó por un enfoque basado en campos de Markov para el modelado de la solución. Una estrategia bayesiana-markoviana permite formalizar las interacciones entre las distintas fuentes de información asociadas a un mismo problema, y puede usarse sistemáticamente para abordar distintas tareas de visión. El algoritmo propuesto, que consiste en la estimación simultánea de un campo de segmentación caracterizado por un modelo de movimiento lineal simplificado, y de un campo auxiliar de líneas de discontinuidad de movimiento, explotó estas ventajas incorporando en el modelo variada información de bajo nivel (campo de movimiento, intensidad de la imagen, medidas parciales de confianza), y especificando sus interacciones mediante la definición de funciones de

energía a minimizar. La optimización de la solución se llevó a cabo mediante un algoritmo de relajación determinística.

Los resultados obtenidos con el método de segmentación propuesto son de buena calidad para las distintas secuencias analizadas, pero no dejan de verse afectados por la interdependencia estimación-segmentación de movimiento, situación inherente al problema mismo de la medición del movimiento. En vista de esto, se considera que, dentro del marco planteado en este trabajo, se pueden conseguir mejoras en el desempeño del algoritmo. Variadas son las rutas que pueden explorarse para optimizarlo y/o extenderlo, y las aplicaciones potenciales.

Finalmente, se concluye que todos los objetivos planteados para este trabajo se cubrieron satisfactoriamente.

Para terminar, hay que decir que el trabajo en este campo es amplio y atractivo, por la variedad de problemas teóricos existentes, de aplicaciones actuales y futuras, y por el interés creciente por abordar, resolver y desarrollar ambos aspectos.

## Apéndice A

### Configuraciones de los cliques del campo de líneas.

En la figura 0.1. se muestran las configuraciones y los valores elegidos para los cliques de 4 elementos (primer orden). En la figura 0.2 se muestran los de segundo orden. Las estructuras geométricas utilizadas son similares a las de [GARC95a], pero los valores específicos son diferentes.

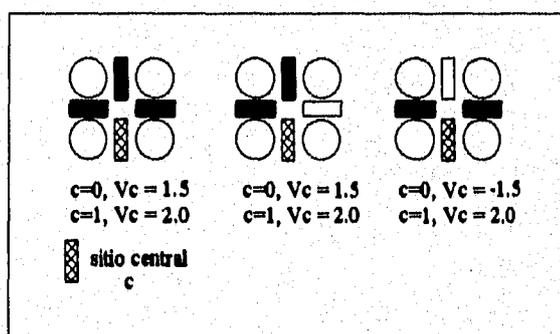


Figura 0.1: Cliques de cuatro elementos del campo de líneas. Otros cliques de cuatro elementos son subconjuntos de los cliques mostrados en la siguiente figura.

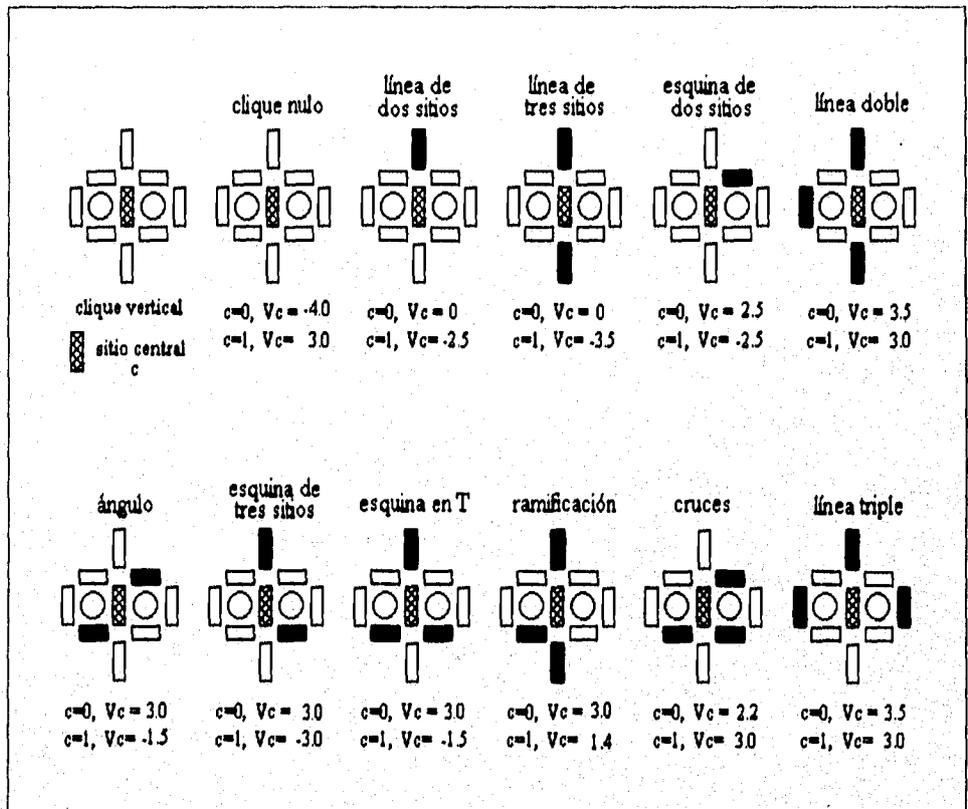


Figura 0.2: Cliques de segundo orden del campo de líneas. Configuraciones analizadas. Se muestra el clique central vertical. Para el clique central horizontal se utilizaron las mismas configuraciones. Las demás configuraciones utilizadas se obtienen por tres operaciones: rotación de 180 grados con respecto al eje perpendicular a la página, reflexión con respecto al eje vertical del clique, y operaciones sucesivas de rotación y reflexión.

## Referencias Bibliográficas.

- [ADIV85]. G. Adiv. Determining three-dimensional motion and structure from optical flow generated by several moving objects. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. PAMI-7, No. 4, julio de 1985. pp. 384-401.
- [AGGA88]. J.K. Aggarwal y N. Nandhakumar. On the computation of motion from sequences of images. Proceedings of the IEEE. Vol. 76, No. 8, agosto de 1988.
- [AIZA95]. K. Aizawa y T.S. Huang. Model-based image coding: advanced video coding techniques for very low bit-rate applications. Proceedings of the IEEE. Vol. 83, No. 2, febrero de 1995.
- [ANAN93]. P. Anandan et al. Hierarchical model-based motion estimation. En "Motion Analysis and Image Sequence Processing". M.I Sezan y R.L. Lagendijk, eds.. Norwell, MA, Kluwer, 1993.
- [BAAZ91a]. N. Baaziz y C. Labit, Multigrid motion estimation on pyramidal representations for image sequence coding. Reporte tecnico 572, IRISA - Rennes, febrero de 1991.
- [BAAZ91b]. N. Baaziz, Approches d'estimation et de compensation de mouvement multiresolutions pour le codage de sequences d'images. Tesis Doctoral, Universidad de Rennes I, Francia, octubre de 1991.
- [BAAZ94a]. N. Baaziz y C. Labit. Wavelet transform and motion estimation for image sequence coding: a multiconstraint approach. En "Wavelets in image communication". M. Barlaud, Ed. Amsterdam, Elsevier Science B.V., 1994.
- [BAAZ94b]. N. Baaziz y C. Labit. Multiconstraint Wiener-based motion compensation using wavelet pyramids. IEEE Transactions on Image Processing. Vol. 3, No. 5, septiembre de 1994. pp 688-692.
- [BARR94]. J. L. Barron, D. J. Fleet, y S. S. Beauchemin. Systems and Experiment. Performance of Optical Flow Techniques. International Journal of Computer Vision. Vol. 12. No. 1. 1994. pp 43-77.

- [**BEAL88**]. E. M. L. Beale. Introduction to Optimization. New York, John Wiley and Sons, 1988.
- [**BERT88**]. M. Bertero, T. Poggio y V. Torre. Ill-posed problems in early vision. Proceedings of the IEEE. Vol. 76, No. 8, agosto de 1988, pp 869-889.
- [**BIEM87**]. J. Biemond, L. Looijenga D. E. Boeke and R. Plompen. A pel-recursive Wiener-based displacement estimation algorithm. Signal processing. Vol. 13, No. 4, diciembre de 1987. pp 399-412.
- [**BIER86**]. M. Bierling, y R. Thoma. Motion compensating field interpolation using a hierarchically structured displacement estimator. Signal Processing. Vol. 11. 1986. pp. 387-404.
- [**BRAI95a**]. J.C. Brailean y A.K. Katsaggelos. A recursive nonstationary MAP displacement vector field estimation algorithm. IEEE Transactions on Image Processing. Vol. 4, No. 4, abril de 1995. pp 416-429.
- [**BRAI95b**]. J.C. Brailean et al. Noise reduction filters for dynamic image sequences: a review. Proceedings of the IEEE. Vol. 83, No. 9, septiembre de 1995.
- [**CAFF76**]. C. Cafforio y F. Rocca. Methods for measuring small displacements of television images. IEEE Transactions on Information Theory. Vol. IT-22, No. 5, septiembre de 1976. pp. 573-579.
- [**CHAN93**]. M. M. Chang, A. M. Tekalp y M. I. Sezan. Motion-field segmentation using an adaptive MAP criterion. Proceedings of the IEEE International Conference on ASSP, 1993.
- [**CHEL93**]. R. Chellapa y A.K. Jain, eds. Markov random fields. Theory and Application. San Diego, Academic Press, 1993.
- [**CHEO93**]. C. K. Cheong, K. Aizawa, T. Saito y M. Hatori. Motion estimation with wavelet transform and the application to motion compensated interpolation. Proceedings of the ICASSP 93. IEEE. Vol. 5. pp 217-220. 1993.
- [**CHIA95**]. L. Chiariglione. The development of an integrated audiovisual coding standard: MPEG. Proceedings of the IEEE, Vol. 83, No. 2, Febrero 1995.
- [**DUBO93**]. E. Dubois y J. Konrad. Estimation of 2-D motion fields from image sequences with application to motion compensated processing. En "Motion analysis and image sequence processing ". M.I Sezan y R.L. Lagendijk, eds. Norwell, MA., Kluwer Academic Publishers. 1993.
- [**FRAN91**]. E. Francois. Interpretation qualitative du mouvement a partir d'une sequence d'images. Tesis Doctoral, Universidad de Rennes I, Francia, junio de 1991.

- [**GARC95a**]. V. García Garduño, Une approche de compression orientee-objets par suivi de segmentation basee mouvement pour le codage de sequences d'images numeriques. Tesis Doctoral, Universidad de Rennes I, Francia, mayo de 1995.
- [**GARC95b**]. V. García Garduño, Notas del curso "Estimación y compensación de movimiento en secuencias de imagenes". DEPFI-UNAM, 1995.
- [**GATI96**]. D. Gatica Pérez. Estimación de movimiento y segmentación basada en movimiento en secuencias de imágenes. Reporte técnico. DEPFI-UNAM, 1996.
- [**GEMA84**]. S. Geman y D. Geman. Stochastic relaxation, Gibbs distributions, and the bayesian restoration of images. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. PAMI-6, No. 6, noviembre de 1984. pp. 721-741.
- [**HARA92**]. R. M. Haralick, y L. G. Shapiro, Computer and Robot Vision. Reading, Mass., Addison-Wesley Publishing Co., 1992.
- [**HAYK89**]. S. Haykin. Digital filters. Macmillan Publishing Co., 1989.
- [**HILD87**]. E. Hildreth. Computations underlying the measurement of visual motion. En "Image Understanding 1985-1986". W. Richards y S. Ullman, Eds. Ablex Publishing Co, 1987.
- [**HEIT93**]. F. Heitz y P. Bouthemy. Multimodal estimation of discontinuous optical flow using Markov Random Fields. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 15, No. 12, diciembre de 1993. pp. 1217-1232.
- [**HORN86**]. B.K.P. Horn. Robot vision. The MIT Press - McGraw Hill Co. Cambridge, Ma. Septima reimpression, 1991.
- [**HOTT88**]. M. Hotter y R. Thoma. Image segmentation based on object oriented mapping parameter estimation. Signal Processing. Vol. 15. 1988. pp. 315-334.
- [**HUAN81**]. T.S. Huang, ed. Image Sequence Analysis. Berlin, Springer-Verlag, 1981.
- [**HUAN83**]. T.S. Huang, ed. Dynamic Scene Analysis and Image Sequence Processing. Berlin, Springer-Verlag, 1983.
- [**HUAN92**]. T.S. Huang. Visual Motion Analysis. en "Encyclopedia of Artificial Intelligence". S. Shapiro, ed. Segunda Edicion. Nueva York, John Wiley and Sons, 1992.

- [**HUAN93**]. J. Huang y R. M. Mersereau. Contour-based hybrid displacement estimation for image sequence compression. Proceedings of the ICASSP 93. IEEE. Vol. 5. pp 433-436. 1993.
- [**JAIN89**]. A.K. Jain. Fundamentals of digital image processing. New Jersey, Prentice-Hall, 1989.
- [**KEAR87**]. J. K. Kearney, W. B. Thompson y D. L. Boley. Optical flow estimation: an error analysis of gradient-based methods with local optimization. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 9, No. 2, marzo de 1992. pp. 229-244.
- [**KONR92**]. J. Konrad y E. Dubois. Bayesian estimation of motion vector fields. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 14, No. 9, septiembre de 1992. pp. 910-927.
- [**LI93**]. W. Li. y F. X. Mateo. Segmentation based coding of motion compensated prediction error images. Proceedings of the ICASSP 93. IEEE. Vol. 5. pp 357-360. 1993.
- [**LI94**]. H. Li, A. Lundmark y R. Forchheimer. Image sequence coding at very low bitrates: a review. IEEE Transactions on image processing. Vol. 3, No. 5, septiembre de 1994. pp. 589-609.
- [**LI95**]. S. Z. Li. Markov Random Field Modeling in Computer Vision. Tokio, Springer-Verlag, 1995.
- [**LIMB75**]. J. O. Limb y J. A. Murphy. Measuring the speed of moving objects from television signals. IEEE Transactions on Communications. Vol., No. , abril de 1975. pp. 474-478.
- [**MARR82**]. D. Marr. Vision: A computational study into the human representation and processing of visual information. W. H. Freeman & Co., San Francisco, 1982.
- [**MARR85**]. J.L. Marroquin. Probabilistic solution of inverse problems. Tesis Doctoral. MIT, Cambridge, septiembre de 1985.
- [**MOOR87**]. R.J. Moorhead II, S.A. Rajala y L.W. Cook. Image sequence compression using a pel-recursive motion-compensated technique. IEEE Journal on Selected Areas in Communications. Vol. SAC-5, No. 7, agosto de 1987. pp. 1100-1114.
- [**MURR87**]. D.W. Murray, y B.F. Buxton. Scene segmentation from visual motion using global optimization. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. PAMI-9, No. 2, marzo de 1987. pp. 220-228.

- [**MURR90**]. D.W. Murray, y B.F. Buxton. Experiments on the machine interpretation of visual motion. Cambridge, The MIT Press, 1990.
- [**MUSM85**]. H.G. Musmann, P. Pirsch y H.J. Grallert. Advances in picture coding. Proceedings of the IEEE . Vol. 73, No. 4, abril de 1985. pp. 572-597.
- [**NAGE86**]. H. H. Nagel y W. Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. PAMI-8, No. 5, septiembre de 1986. pp. 565-593.
- [**NETR79**]. A.N. Netravali y J.D. Robbins. Motion-compensated television coding: part I. The Bell System Technical Journal. Vol. 58, No. 3, marzo de 1979. pp. 631-670.
- [**NETR80a**]. A.N. Netravali y J.O. Limb. Picture coding: a review. Proceedings of the IEEE. Vol. 68, No. 3, marzo de 1980. pp. 366-406.
- [**NETR80b**]. A.N. Netravali y J.D. Robbins. Motion-compensated coding: some new results. The Bell System Technical Journal. Vol. 59, No. 9, noviembre de 1980. pp. 1735-1745.
- [**NETR88**]. A. N. Netravali y B.G. Haskell. Digital pictures: representation and compression. Plenum Press, 1988.
- [**NICO91**]. H. Nicolas y C. Labit. Global motion identification for image sequence analysis and coding. Proceedings of the ICASSP 91, IEEE. pp 2825-2828.
- [**NICO92a**]. H. Nicolas, y C. Labit, Region-based motion estimation using deterministic relaxation schemes for image sequence coding. Proceedings of the ICASSP 92, IEEE. Vol. 3. pp 265-269.
- [**NICO92b**]. H. Nicolas. Hierarchie de modeles de mouvement et methodes d'estimation asociees. Application au codage de sequences d'images. Tesis Doctoral, Universidad de Rennes I, Francia, septiembre de 1992.
- [**PAPP92**]. T.N. Pappas. An adaptive clustering algorithm for image segmentation. IEEE Transactions on Signal Processing. Vol. SP-40, abril de 1992, pp. 901-914.
- [**PAQU83**]. R. Paquin y E. Dubois. A spatio-temporal gradient method for estimating the displacement field in time-varying imagery. Computer Vision, Graphics and Image Processing. Vol. 21, 1983. pp 205-221.
- [**POTT75**]. J.L. Potter, Velocity as a cue to segmentation. IEEE Transactions on Systems, Man and Cybernetics, mayo de 1975.

- [**SABR84**]. S. Sabri, Movement compensated interframe prediction for NTSC color TV signals. *IEEE Transactions on Communications*. Vol. COM-32, No. 8, agosto de 1984. pp. 954-968.
- [**SEZA93**]. M.I Sezan y R.L. Lagendijk, eds. *Motion analysis and image sequence processing*. Norwell, MA., Kluwer Academic Publishers. 1993.
- [**STIL91**]. C. Stiller y D. Lappe. Gain/cost controled displacement-estimation for image sequence coding. *Proceedings of the ICASSP 91, IEEE*. pp 2729-2732.
- [**STIL93**]. C. Stiller. A statistical image model for motion estimation. *Proceedings of the ICASSP 93, IEEE*. Vol. 5. pp 193-196.
- [**TEKA95**]. A. M. Tekalp. *Digital video processing*. New Jersey, Prentice Hall, 1995.
- [**TZIR88**]. G. Tziritas. Estimation recursive-iterative du mouvement apparent dans une sequence d'images. *Traitement du Signal*. Vol. 5. No. 4, 1988. pp. 305- 310.
- [**VERR89**]. A. Verri y T. Poggio. Motion field and optical flow: qualitative properties. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 11, No. 5, mayo de 1989. pp. 490-498.
- [**WALK84**]. D.R. Walker and K.R. Rao. Improved pel-recursive motion compensation. *IEEE Transactions on Communications*. Vol. COM-32, No. 10, octubre de 1984. pp. 1128-1134.
- [**WALK87**]. D.R. Walker y K.R. Rao. Motion-compensated coder. *IEEE Transactions on Communications*. Vol. COM-35, No. 11, noviembre de 1987. pp. 1171-1178.
- [**WANG92**]. Q. Wang y R.J. Clarke. Motion estimation and compensation for image sequence coding. *Signal Processing: Image Communication*. Vol. 4, No. 2, abril de 1992. pp 161-174.
- [**WANG93**]. J. Y. A. Wang y E. H. Adelson. Layered representation for image sequence coding. *Proceedings of the ICASSP 93, IEEE*. Vol. 5. pp 221-224. 1993.
- [**ZHEN95**]. H. Zheng. Estimation of image motion in scenes containing multiple moving objects. Tesis Doctoral. Universidad de Queen. Kingston, Ontario, Canada, abril de 1995.