

Tema 5: Procesamiento de vídeo

5.1. Introducción

5.2. Detección y estimación de movimiento

5.3. Filtrado de secuencias de vídeo

5.4. Segmentación

5.5. Análisis de secuencias

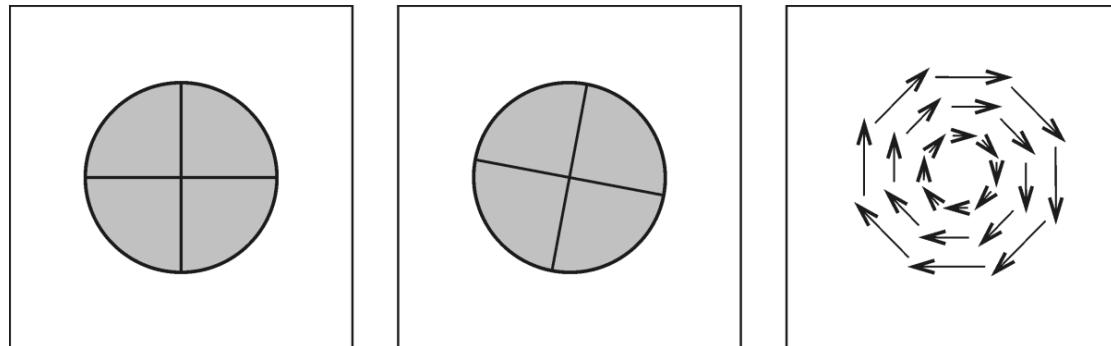
Fundamentos

- **Detección de movimiento (DM)**: identificar qué regiones de una escena se han movido.
- **Estimación de movimiento (EM)**: medir cómo se mueven dichas regiones.
- El movimiento de un objeto se describe generalmente mediante un **Vector de Movimiento VM** → podemos extraer magnitud y dirección del movimiento.
- Muchos algoritmos de EM en 2D están basados en el concepto de **flujo óptico**. **Flujo óptico**: patrón de *movimiento aparente* de objetos, superficies y bordes en una escena → puede ser calculado midiendo variaciones de niveles de intensidad en el tiempo.
- **Campo de flujo óptico**: contiene los VM de cada región (pixel, bloque o región) de un *frame* → aproximación del movimiento 3D de los objetos en un *frame* 2D.

Fundamentos

Los cálculos para la obtención del flujo óptico se basan en dos asunciones:

1. La intensidad observada en un objeto es constante en el tiempo.
2. Los puntos cercanos en el espacio tienden a moverse de manera similar.



Dos frames en diferentes instantes (t_1 y $t_1 + \Delta t$) y el flujo óptico resultante.

From [Marques11]

Fundamentos

- El problema de EM es complejo:

- La escena real es 3D, pero los algoritmos de EM tienen acceso únicamente a una representación 2D → deben confiar en el **movimiento aparente** de los objetos relativo al *frame*, que no siempre corresponde con el movimiento real en la escena original.
- Los algoritmos de EM deben trabajar al **nivel de granularidad apropiado** para una tarea determinada → obtener un VM para cada pixel no es generalmente buena idea; tampoco obtener un VM para todo el *frame* → compromiso, generalmente se divide el *frame* en bloques, un VM por bloque.

Fundamentos

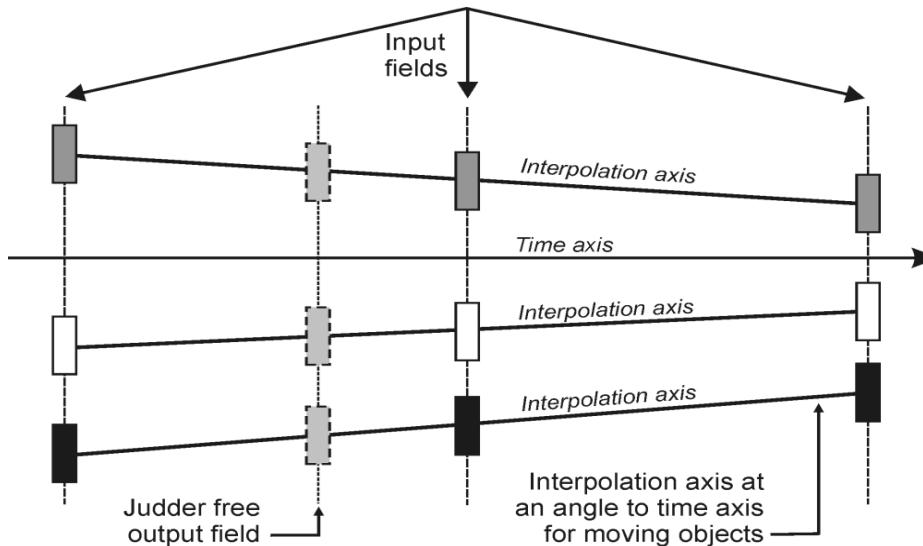
• El problema de EM es complejo:

- En algunos casos, p.e. seguimiento de objetos, queremos calcular el movimiento de objetos en una escena, necesitamos que estén accesibles como entidades individuales → requiere **segmentación de objetos previa a la EM**.
- Los métodos de EM son fuertemente dependientes de la aplicación:
 - **Compresión:** el objetivo es generar VM que permitan al decodificador recrear el *frame* basándose en las diferencias entre el *frame* actual y el anterior, ahorrando bits en el proceso.
 - En los estándares de compensación de movimiento el objetivo es **alinear el eje de interpolación con la dirección del movimiento** de los objetos a través de los *frames*.

Fundamentos

Compensación de movimiento (CM): proceso por el cual los resultados de la EM son empleados para mejorar el desempeño de los algoritmos de procesado de vídeo.

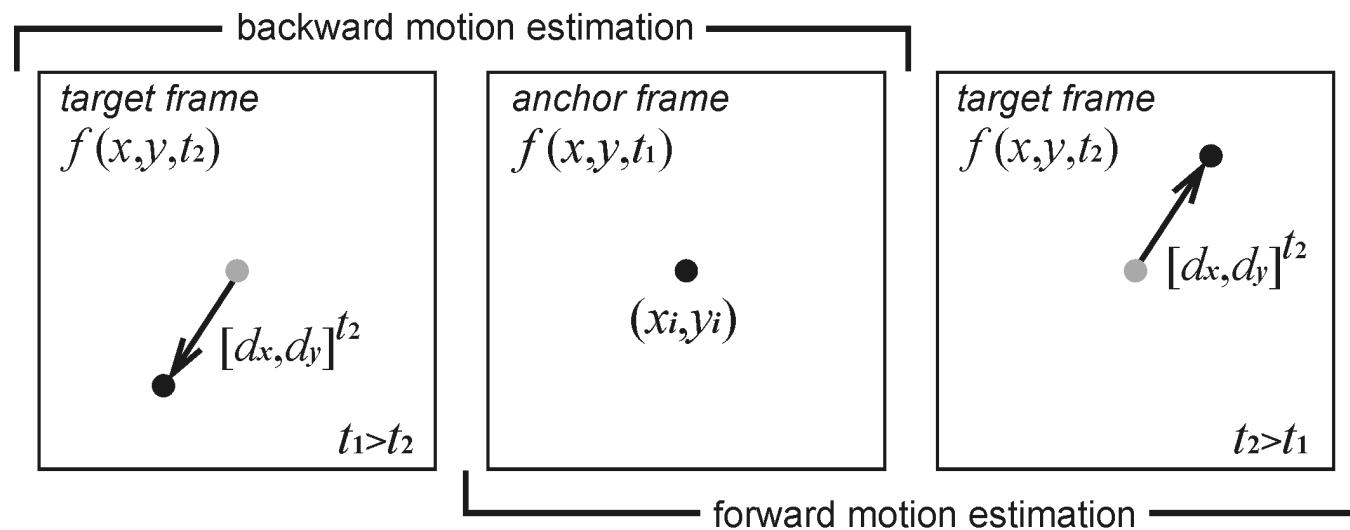
- Los resultados de la EM son utilizados para crear **un eje de interpolación equivalente alineado con la trayectoria de cada objeto en movimiento.**



From [Marques11]

Métodos de Estimación de Movimiento (EM)

- **Problema:** estimar el movimiento entre dos *frames* $f(x,y,t_1)$ (referencia) y $f(x,y,t_2)$ (objetivo).
- El VM $[d_x d_y]^T$ representa el **desplazamiento de un punto** (x_i, y_i) entre t_1 y t_2 .



From [Marques11]



Métodos de Estimación de Movimiento (EM)

- **Métodos basados en características:** establecen una correspondencia entre pares de características seleccionadas en los dos *frames* e intentan ajustar un modelo de movimiento utilizando **métodos como el ajuste por mínimos cuadrados**.
- **Métodos basados en intensidad:** se basan en la asunción de intensidad constante y abordan en problema de EM como un problema **de optimización**:
 - ¿cómo representar el movimiento?
 - ¿qué criterios utilizar para estimar los parámetros de movimiento?
 - ¿cómo buscar los parámetros de movimiento óptimos?

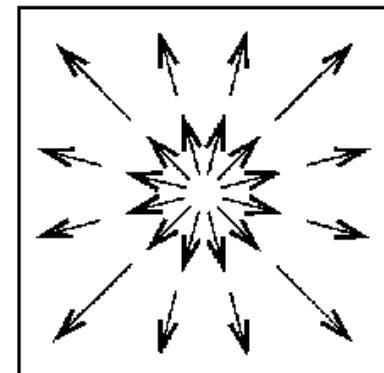
Representación de Movimiento (EM)

a) Global: un único modelo de movimiento para todo el *frame*.

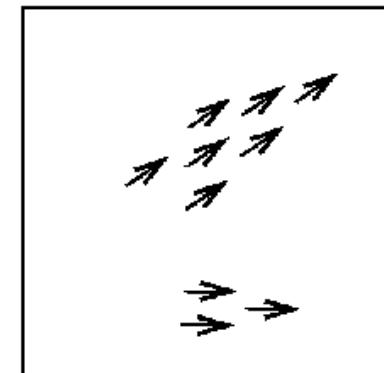
b) Basada en pixel: un VM por pixel.
Alto coste computacional.

c) Basada en bloque: encontrar el VM que mejor representa el movimiento asociado a cada bloque (H.261, H.263, MPEG1, MPEG2).

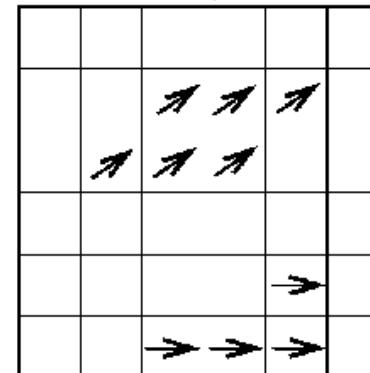
d) Basada en regiones: el *frame* se divide en regiones que corresponden con un objeto con movimiento consistente (MPEG4).



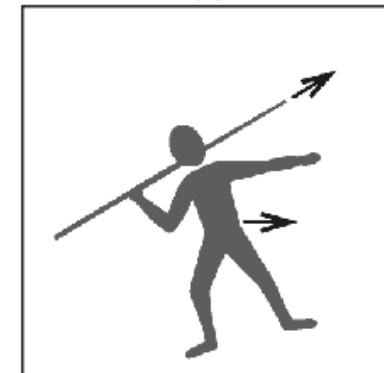
(a)



(b)



(c)



(d)

From [Marques11]

Criterio de estimación de movimiento

Displace frame difference (DFD): calcular las diferencias de intensidad entre cada punto del *frame* de referencia y el *frame* objetivo.

$$E_{DFD}(d) = \sum_{n \in R} (f_2(n+d) - f_1(n))^p$$

- $f_1(\mathbf{n})$: valor del pixel \mathbf{n} en t_1 .
- $f_2(\mathbf{n}+\mathbf{d})$: valor del pixel $\mathbf{n}+\mathbf{d}$ en t_2 ; \mathbf{d} es el desplazamiento estimado.
- R es la región considerada.
- Para $p=1 \rightarrow$ Media diferencia absoluta (Mean Absolute Difference MAD)
- Para $p=2 \rightarrow$ Media Error cuadrático (Mean Square Error MSE).

Métodos de optimización

- **Búsqueda exhaustiva:** busca a través de todas las posibles combinaciones de valores → alcanza un **mínimo global** óptimo a expensas de un **alto coste computacional** → se suele utilizar con AD.
- **Búsqueda basada en gradiente:** utiliza métodos numéricos de cálculo de gradiente → alcanza un punto **óptimo local próximo** a la solución inicial → se suele utilizar con SE.
- **Búsqueda de resolución múltiple:** comienza la búsqueda a una **baja resolución** y va propagando las soluciones parciales hacia **resoluciones más altas** para que puedan ser refinadas → buen compromiso entre velocidad y mínimos locales.

Algoritmos de EM

Block Matching Algorithms (BMA)

Ideas básicas

- *Frame* se divide en M bloques nos solapados.
- El movimiento de cada bloque se asume constante.
- El bloque entero puede ser codificado por un VM.

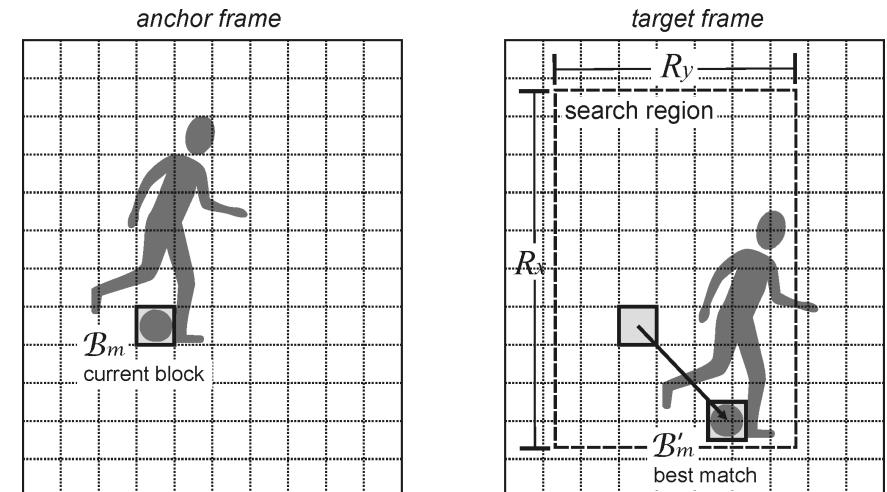
Problema. Encontrar el mejor VM para cada bloque.

Algoritmos

- *Exhaustive Search Block Matching Algorithm (EBMA)*
- *Fast Algorithms*
 - *2D Log Search Method*
 - *Three-Step Search Method*
 - *Hierarchical Block Matching Algorithm (HBMA)*
- *Phase Correlation Method*

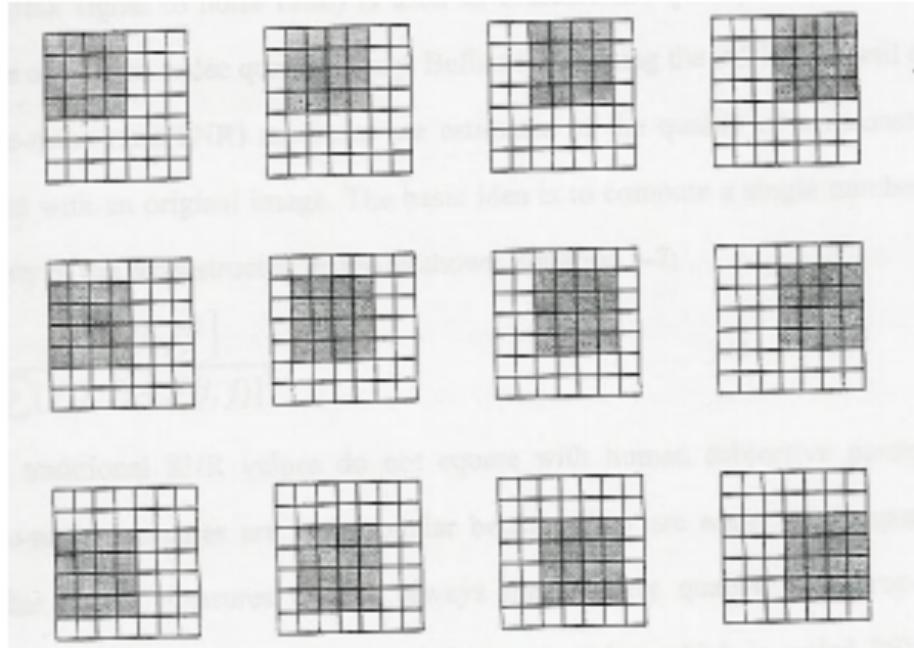
Exhaustive Search Block Matching Algorithm (EBMA)

- Compara cada bloque en el *frame* de referencia con todos los bloques (dentro de un rango) en el *frame* objetivo → selecciona el que optimiza una función de coste como AD.
- Debilidades:
 - Los VM se estiman de forma Independiente de bloque a bloque.
 - Artefactos de bloque.



From [Marques11]

Exhaustive Search Block Matching Algorithm (EBMA)



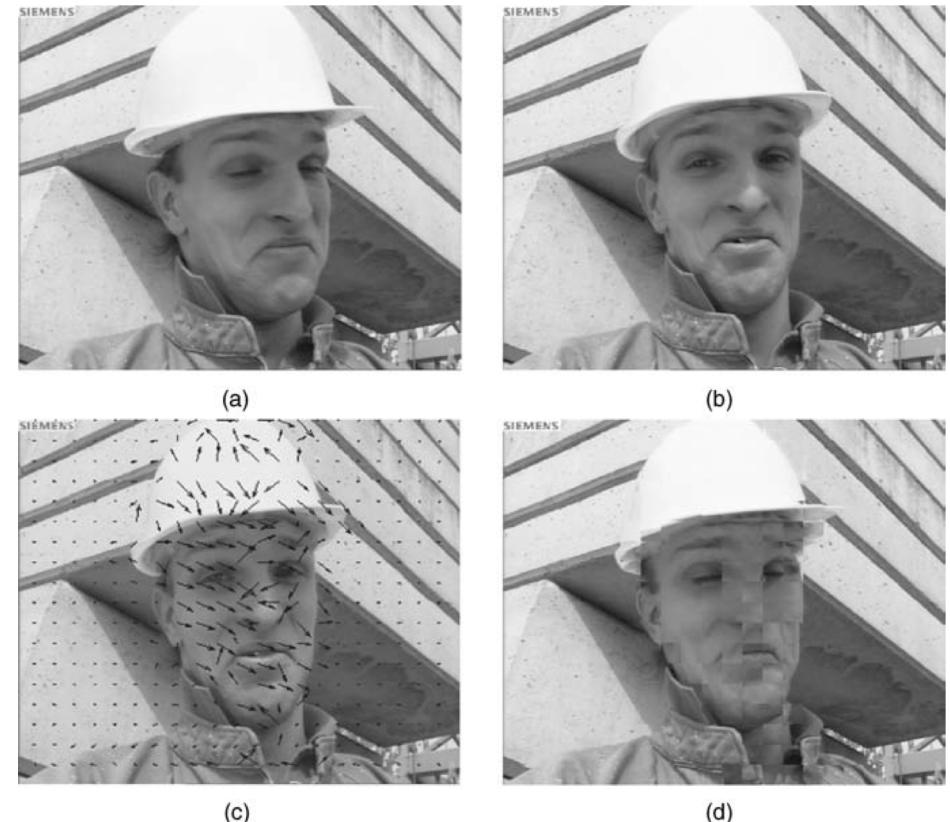
512x512 frame with N=P=16 the number of operation per frame is 2.85×10^8 so with a frame rate of 30fps the number will be 8.55×10^9 per minute

El número de candidatos es: $(2p+1)^2$

Exhaustive Search Block Matching Algorithm (EBMA)



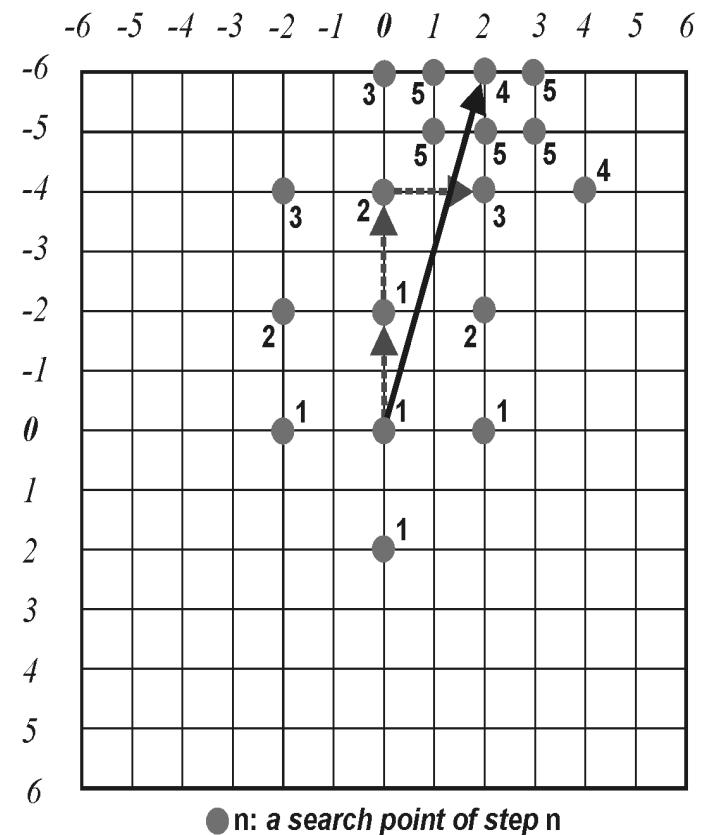
- a) *Frame* estimado.
- b) *Frame* de referencia.
- c) Campo de movimiento sobrepuuesto sobre el *frame* de referencia.
- d) *Frame* reconstruido.



From [Marques11]

Fast Algorithms. 2D Log Search Method

- Algoritmo rápido de búsqueda basada en gradiente.
- Comienza por la posición correspondiente a desplazamiento cero y prueba **cinco posiciones** cercanas en forma de diamante.
- Selecciona el bloque que proporciona el **menor error** y lo utiliza para realizar una **nueva búsqueda** a su alrededor.
- Si la mejor opción es el **bloque central**, procede con una **nueva región de búsqueda** cuyo **desplazamiento es la mitad que el desplazamiento previo**.
- El proceso se repite hasta que el **desplazamiento es 1**.



Fast Algorithms. 2D Log Search Method

Frame 0 Image



Frame 1 Image



Image Size 243 x 360

Motion Vectors

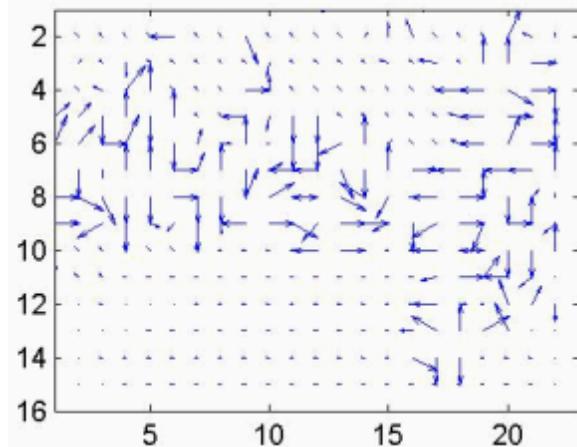
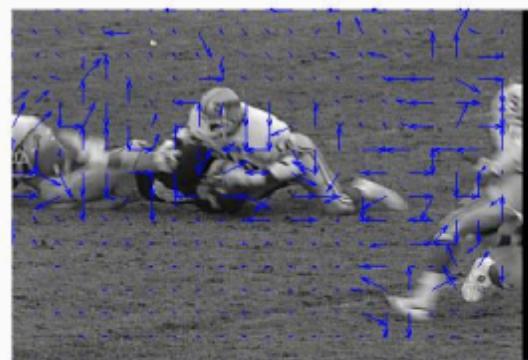


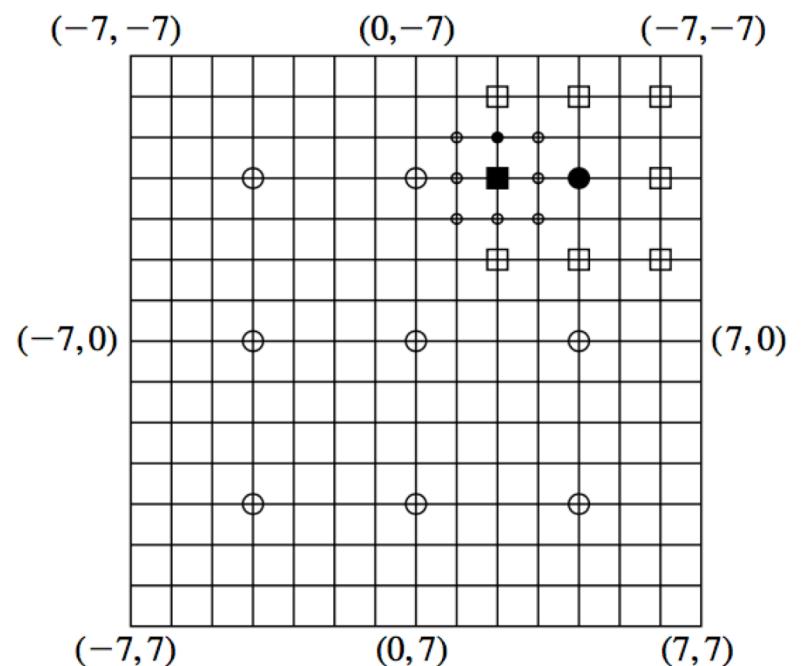
Image Size 243 x 360

Motion Vectors



Three-Step Search Method

- Comienza por la posición correspondiente a desplazamiento cero y prueba **nueve posiciones** cercanas, con un desplazamiento **de $p/2$** , donde p es **la mitad del rango de búsqueda en cada dirección (horizontal y vertical)**.
- Selecciona el bloque que proporciona el **menor error** y lo utiliza para realizar una nueva búsqueda a su alrededor.
- La **nueva** región de búsqueda tiene un desplazamiento que es la **mitad que el desplazamiento previo**.
- El proceso se repite hasta que el **desplazamiento es 1**.



Three-Step Search Method

Frame 0 Image



Frame 1 Image



Image Size 243 x 360

Motion Vectors

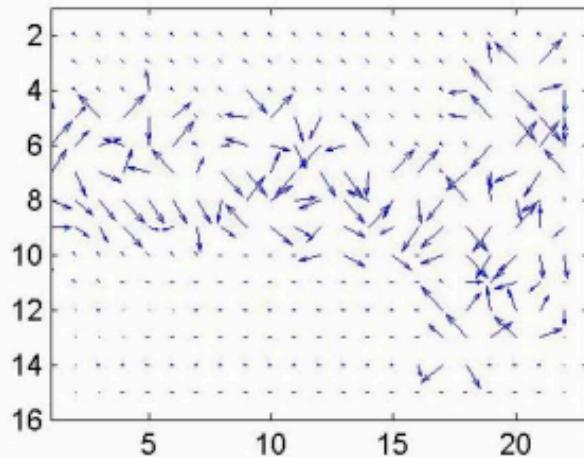
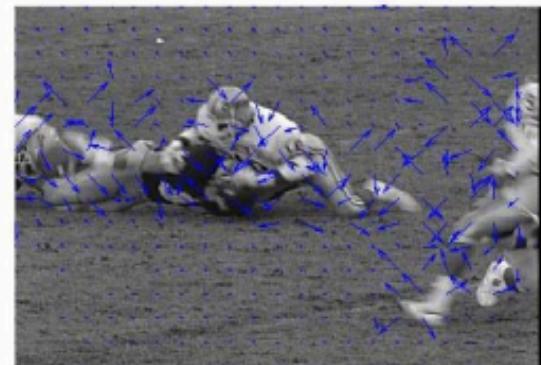


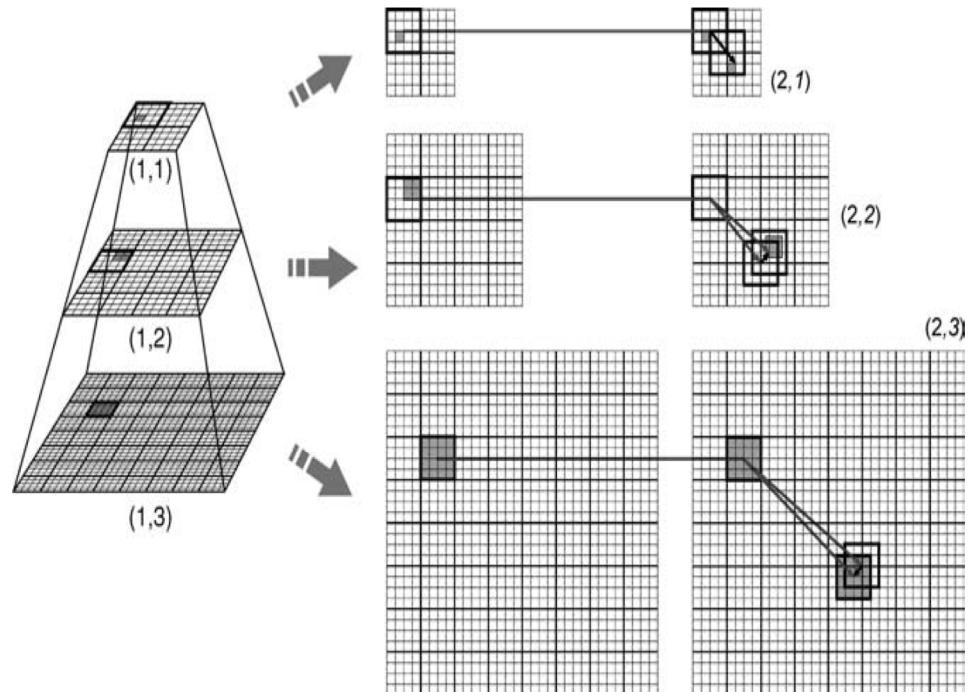
Image Size 243 x 360

Motion Vectors



Hierarchical Block Matching Algorithm (HBMA)

- Es un algoritmo de resolución múltiple.
- Primero estima los **VM en un par de frames de baja resolución** (filtrados paso bajo y submuestreados).
- Refina la solución inicial utilizando sucesivamente **mayor resolución** y **menor área de búsqueda**.
- Generalmente estructura piramidal, la **resolución se reduce a la mitad en ambas dimensiones en sucesivos niveles**.
- Compromiso entre tiempo de ejecución y calidad de los resultados.



Phase correlation Method

EM en el dominio frecuencial.

1. Calcular la DFT de cada bloque de los *frames* referencia (k) y objetivo (k+1).

2. Calcular la potencia espectral cruzada normalizada:

$$C_{k,k+1} = \frac{T_{k+1}T_k^*}{|T_{k+1}T_k^*|}$$

3. Calcular la función de correlación de fase:

$$PFC(n) = T^{-1}(C_{k,k+1}) = \delta(n + d)$$

4. Localizar los picos de PFC → corresponden con las posiciones asociadas a movimiento.

Limitación. Falta de correspondencia entre picos y objetos en movimiento en el *frame*.

Bibliografía

[Marques11] Practical Image and Video Processing Using Matlab. Oge Marques. Wiley-IEEE Press, 2011. Chapter 22: Digital Video Processing Techniques and Applications.

[Bovik09] The Essential Guide to Video Processing. Bovik, Alan C. Academic Press, 2009. Chapter 3: Motion Detection and Estimation.