

TAMAÑO Y LOCALIZACION DE UN OBJETO: UNA APROXIMACION NEURONAL

A. Fernández Caballero¹, Modesto Montoya, Miguel A. Fernández¹ y Francisco J. Gómez¹
Departamento de Informática, Escuela Politécnica Superior,
Universidad de Castilla-La Mancha, 02071 – Albacete, España
e_mail: ¹{caballer, miki, fgomez}@info-ab.uclm.es

RESUMEN DEL TRABAJO. El análisis del movimiento en secuencias de imágenes es una disciplina en constante crecimiento debido al gran número de aplicaciones en el que juega un papel primordial. La presentación de un sencillo modo de obtención, en tiempo real y de una forma neuronal, de dos parámetros fundamentales asociados a los objetos en movimiento (su tamaño y su localización) es el objetivo de este artículo.

INTRODUCCION

El análisis del movimiento en imágenes está creciendo en importancia en numerosas aplicaciones. Algunas de estas aplicaciones son: (a) la televisión en cuanto a la codificación de las imágenes de vídeo mediante compensación, (b) la robótica móvil, (c) el tratamiento de las imágenes de satélite, (d) las aplicaciones, tanto civiles como militares, de seguimiento de objetivos y navegación autónoma, (e) el tratamiento de las imágenes biológicas y médicas, (f) la vigilancia de lugares y su supervisión, y, (g) interfaces y realidad virtual [Mit96]. El problema de la detección de movimiento es particularmente interesante cuando el objetivo a que se persigue es el de la localización espacial de los objetos móviles en la escena. Esta detección de movimiento siempre está fuertemente ligada la detección de cambios temporales en la imagen. Cuando existen objetos en movimiento en una escena, siempre tendremos cambios en la intensidad de los píxeles de la imagen. Este hecho ha dado lugar a una extensa bibliografía, en la que destacamos algunos trabajos ya clásicos con diferentes aproximaciones. Para un estudio más extenso del tema, se recomienda la lectura de [Fer93] [Fer97b]. A destacar igualmente la emergente aproximación de Fernández y Mira [Fer92] [Fer95] [Fer97a] en este sentido.

El seguimiento de elementos de una imagen a otra es un procedimiento común, sobre todo en aplicaciones de vigilancia. Algunos procesos de seguimiento han sido definidos por medio de: (a) un modelo representativo de elementos (por ejemplo, las coordenadas de imagen de algunos puntos característicos, la longitud y la orientación de los segmentos del contorno), (b) un modelo cinético de la evolución de los elementos (por ejemplo, una velocidad constante, una aceleración constante,

un movimiento afín), (c) un conjunto de relaciones entre los parámetros del modelo y las medidas de la imagen, y, (d) un filtro temporal para la estimación de los parámetros del modelo a partir de los datos de la imagen.

TAMAÑO Y LOCALIZACION DE UN OBJETO

Uno de los aspectos primordiales de la estructura de un objeto es su tamaño. Conocer el tamaño de un objeto en términos absolutos es útil para el reconocimiento del objeto [Mut86a] [Mut86b]. A su vez, el aspecto predominante para el cálculo de la velocidad de un objeto sería su localización. Un objeto en traslación o dilatación (no en rotación) puede reducirse a efectos de cálculo del movimiento a la obtención de esos dos parámetros: su localización y su tamaño en todo instante de tiempo t .

Nuestra propuesta es que ambas variables pueden obtenerse de un modo sencillo, conocida la silueta S del objeto en el instante t . Para ello vamos a definir, por un lado, el tamaño a partir de la longitud de dos líneas rectas (o cuerdas) determinadas por cuatro puntos conocidos de la superficie del objeto. Los puntos a los que hacemos referencia son (x_1, y_1) , (x_2, y_2) , (x_3, y_3) y (x_4, y_4) , tales que:

$$\begin{aligned}\forall (x,y) \in S(i,j,t), \quad x_1 < x \\ \forall (x,y) \in S(i,j,t), \quad x_2 > x \\ \forall (x,y) \in S(i,j,t), \quad y_3 < y \\ \forall (x,y) \in S(i,j,t), \quad y_4 > y\end{aligned}$$

Dicho de otro modo, los cuatro puntos son:

$$\begin{aligned}(x_1, y_1) &: \text{ punto más a la izquierda del objeto en la imagen} \\ (x_2, y_2) &: \text{ punto más a la derecha del objeto en la imagen} \\ (x_3, y_3) &: \text{ punto más arriba del objeto en la imagen} \\ (x_4, y_4) &: \text{ punto más abajo del objeto en la imagen}\end{aligned}$$

Las dos cuerdas, que denominamos segmentos de línea máximos del objeto, no van a unir los puntos (x_1, y_1) y (x_2, y_2) , (x_3, y_3) y (x_4, y_4) entre sí, respectivamente, sino sus proyecciones $(X_1, 0)$ y $(X_2, 0)$, $(0, Y_3)$ y $(0, Y_4)$, respectivamente, como puede apreciarse en la figura 2.

En cuanto a la localización del objeto, ésta vendrá determinada por un punto único característico del mismo (X_{obj}, Y_{obj}) , a saber la intersección de los dos segmentos

$(X_1, Y_3)(X_2, Y_4)$ y $(X_2, Y_3)(X_1, Y_4)$. Este punto será denominado punto representativo del objeto.

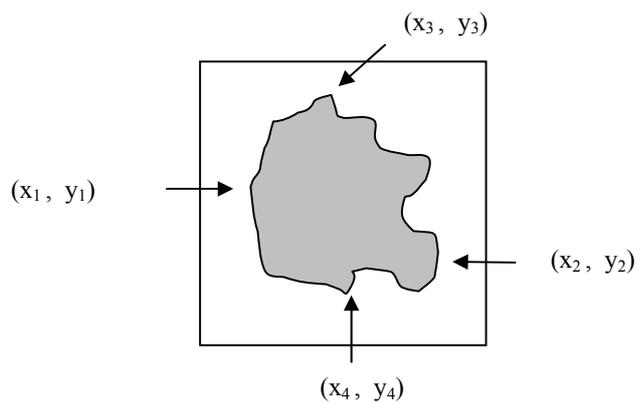


Figura 1. Obtención de los puntos extremos de un objeto.

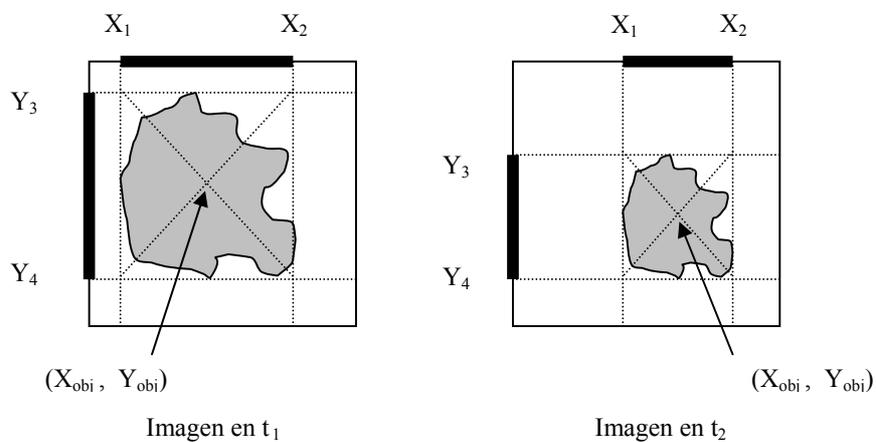


Figura 2. Obtención de los segmentos de línea máximos y del punto representativo del objeto.

IMPLEMENTACION NEURONAL

Se propone una implementación hardware que obtenga los valores X_1 , X_2 , Y_3 e Y_4 en tiempo real. Para ello se propone implementar una estructura neuronal de fácil manejo. Partimos de la estructura básica de neurona multifuncional de la figura 3. En dicha figura tenemos:

- INH_{in} = señal de inhibición procedente de la neurona anterior
- ACT_{in} = señal de activación de entrada de la neurona
- INH_{out} = señal de inhibición hacia la neurona posterior
- ACT_{out} = señal de activación de salida de la neurona

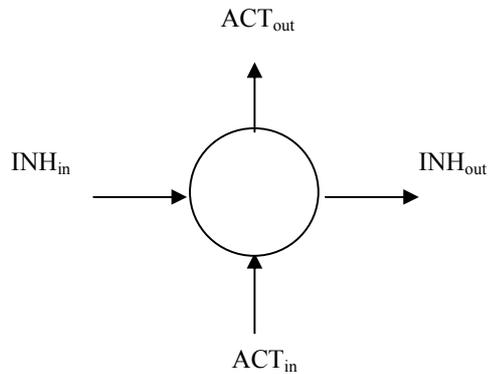


Figura 3. Estructura de la neurona multifuncional.

Esta neurona tiene como característica primaria el poder ser unida en serie con otras neuronas del mismo tipo a través de las señales INH del siguiente modo:

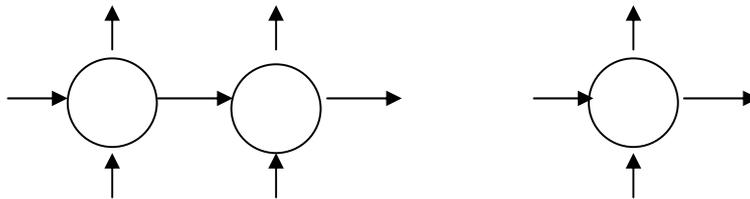


Figura 4. Conexión en serie de las neuronas.

Véase que la señal INH se va propagando con el valor inicial 0 (es decir, no inhibir) hasta una neurona determinada que cumpla la condición adecuada para transmitir a partir de ese momento el valor 1 (inhibir).

En el caso concreto que nos ocupa, se trata de detectar los valores X_1 , X_2 , Y_3 e Y_4 anteriormente descritos. Para ello se propone utilizar cuatro hileras de neuronas dispuestas según la figura 5, donde las señales ACT (tanto ACT_{in} como ACT_{out}) tienen como finalidad el pasar a través de la neurona la información a ser procesada desde un nivel inferior (obtención de la silueta del objeto) a un nivel superior (cálculo de la trayectoria del objeto).

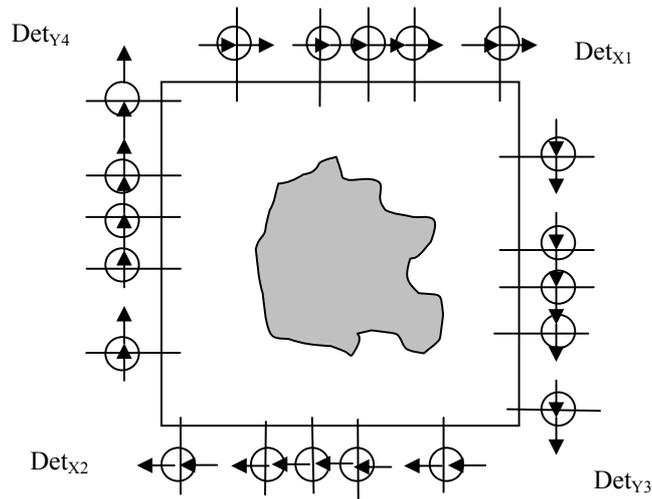


Figura 5. Determinación de los valores X_1 , X_2 , Y_3 , e Y_4 .

El algoritmo se presenta aquí para el caso de las neuronas detectoras de las filas Det_{X1} o Det_{X2} . Para los casos Det_{Y3} o Det_{Y4} , cámbiese i por j , fila por columna, y viceversa.

$$ACT_{in}(i, t) = \begin{cases} 1, & \text{si } \sum S(i, j, t) > 0, j=1..k \\ 0, & \text{en caso contrario} \end{cases} \quad (1)$$

Esta ecuación (1) viene a decir que la neurona de la columna i tiene una activación en su entrada si cualquiera de los pixeles de la imagen de la columna i y filas j está incluido en la silueta del objeto.

$$\text{INH}_{\text{out}}(i, t) = \begin{cases} 1, & \text{si } \text{INH}_{\text{in}}(i, t) = 1 \cup \text{ACT}_{\text{in}}(i, t) = 1 \\ 0, & \text{en caso contrario} \end{cases} \quad (2)$$

Se va propagando a través de la línea de neuronas la señal INH_{out} con valor 0 hasta que ocurre uno de dos posibles eventos (no tienen por qué ser excluyentes entre sí): (a) a la neurona llega un valor de inhibición de 1, o, (b) la neurona recibe una señal de activación desde el nivel inferior. En ambos casos, comienza a propagarse la señal INH_{out} con un valor de 1.

$$\text{ACT}_{\text{out}}(i, t) = i * \text{ACT}_{\text{in}}(i, t) * [1 - \text{INH}_{\text{in}}(i, t)] \quad (3)$$

Esta última ecuación nos ofrece el comportamiento de la neurona en su salida hacia el nivel superior. En este caso la función elegida permite elevar a instancias superiores el valor de la posición de la neurona dentro de la fila.

Vemos como, en cada hilera, a lo sumo una neurona pasará el valor de su posición al nivel superior. Todas las demás valdrán 0. La neurona "ganadora" es, pues, aquella que detecta la primera posición en que se encuentra silueta. Así:

- Det_{X_1} : consigue obtener la posición de la columna donde aparece silueta más a la izquierda, es decir X_1
- Det_{X_2} : consigue obtener la posición de la columna donde aparece silueta más a la derecha, es decir X_2
- Det_{Y_3} : consigue obtener la posición de la columna donde aparece silueta más arriba, es decir Y_3
- Det_{Y_4} : consigue obtener la posición de la columna donde aparece silueta más a bajo, es decir Y_4

PRUEBAS Y RESULTADOS.

Los algoritmos se han aplicado a las secuencias sintéticas SOFA 1, 2 y 3 en un prototipo software programado bajo Microsoft Visual C++. Agradecemos la cortesía de Computer Vision Group, Heriot-Watt University (<http://www.cee.hw.ac.uk/~mtc/sofa>) por el permiso de uso de los fotogramas.

Se muestran en la figura 6 algunos ejemplos de las 20 imágenes que componen cada una de las secuencias. En las tres secuencias segmentamos únicamente el cubo que en ellas aparece, a partir de técnicas estándares, a fin de poder aplicar nuestro algoritmo a la silueta trazada por el mismo. Vemos cómo en la secuencia 1 el cubo está girando sobre sí mismo, en la secuencia 2 se va aproximando al observador, mientras que en la secuencia 3, el cubo se va acercando mientras efectúa una ligera inclinación.

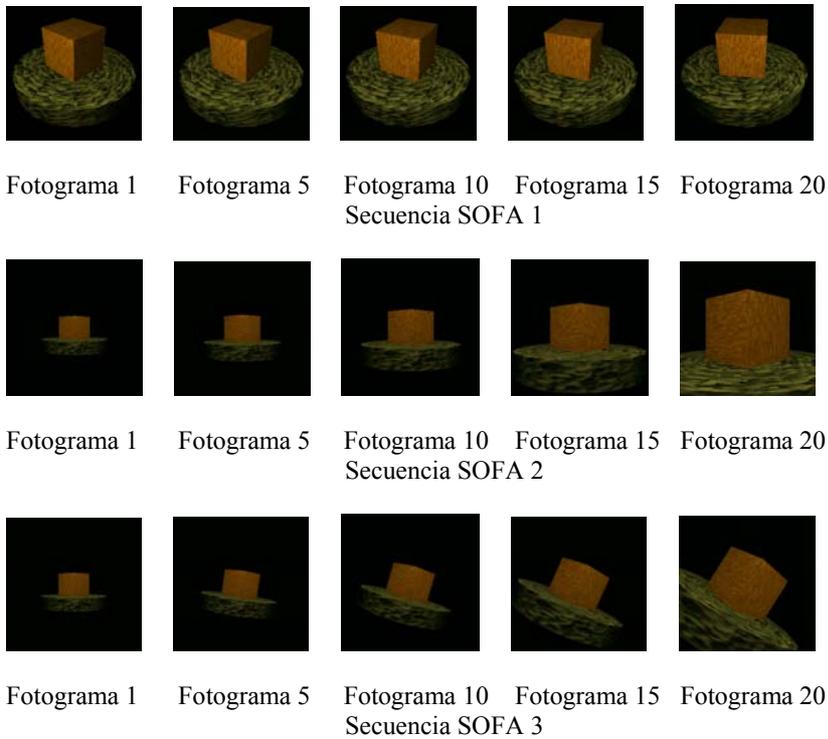


Figura 6. Algunas imágenes de las secuencias SOFA.

Imagen	X ₁	X ₂	Y ₃	Y ₄	X _{obj}	Y _{obj}
s1img1	68	189	23	146	128.5	84.5
s1img2	68	189	23	146	128.5	84.5
s1img3	67	188	23	145	127.5	84
s1img4	67	188	23	145	127.5	84
s1img5	67	187	23	145	127	84
s1img6	67	187	23	145	127	84
s1img7	67	186	23	145	126.5	84
s1img8	67	185	23	145	126	84
s1img9	68	184	23	144	126	83.5
s1img10	68	183	24	144	125.5	84
s1img11	68	182	24	144	125	84
s1img12	69	181	24	143	125	83.5
s1img13	69	180	24	142	124.5	83
s1img14	70	179	24	142	124.5	83
s1img15	71	178	25	141	124.5	83
s1img16	72	177	25	141	124.5	83
s1img17	72	176	25	140	124	82.5
s1img18	73	174	25	139	123.5	82
s1img19	74	173	26	139	123.5	82.5
s1img20	76	172	26	138	124	82

Tabla 1. Resultados para la secuencia SOFA 1.

Imagen	X ₁	X ₂	Y ₃	Y ₄	X _{obj}	Y _{obj}
s2img1	100	157	106	151	128.5	128.5
s2img2	99	158	105	152	128.5	128.5
s2img3	98	159	104	153	128.5	128.5
s2img4	97	160	103	154	128.5	128.5
s2img5	95	162	102	155	128.5	128.5
s2img6	94	163	101	156	128.5	128.5
s2img7	93	164	100	157	128.5	128.5
s2img8	91	166	98	159	128.5	128.5
s2img9	89	168	97	160	128.5	128.5
s2img10	87	170	95	162	128.5	128.5
s2img11	85	172	93	164	128.5	128.5
s2img12	83	174	91	166	128.5	128.5
s2img13	80	177	88	169	128.5	128.5
s2img14	78	179	85	172	128.5	128.5
s2img15	74	183	82	175	128.5	128.5
s2img16	71	186	78	179	128.5	128.5
s2img17	67	190	74	183	128.5	128.5
s2img18	62	195	68	189	128.5	128.5
s2img19	56	201	62	195	128.5	128.5
s2img20	50	207	53	204	128.5	128.5

Tabla 2. Resultados para la secuencia SOFA 2.

Imagen	X ₁	X ₂	Y ₃	Y ₄	X _{obj}	Y _{obj}
s3img1	100	157	106	151	128.5	128.5
s3img2	98	159	106	152	128.5	129
s3img3	96	161	105	152	128.5	128.5
s3img4	95	162	103	154	128.5	128.5
s3img5	92	165	101	156	128.5	128.5
s3img6	90	167	99	158	128.5	128.5
s3img7	88	169	96	161	128.5	128.5
s3img8	86	171	94	163	128.5	128.5
s3img9	83	174	91	166	128.5	128.5
s3img10	81	176	88	169	128.5	128.5
s3img11	78	179	85	172	128.5	128.5
s3img12	74	183	82	175	128.5	128.5
s3img13	71	186	78	179	128.5	128.5
s3img14	67	190	74	183	128.5	128.5
s3img15	63	194	69	188	128.5	128.5
s3img16	58	199	64	193	128.5	128.5
s3img17	53	204	58	199	128.5	128.5
s3img18	47	210	52	205	128.5	128.5
s3img19	40	217	45	212	128.5	128.5
s3img20	32	225	36	221	128.5	128.5

Tabla 3. Resultados para la secuencia SOFA 3.

Los resultados obtenidos para las tres secuencias pueden estudiarse en las tablas 1, 2 y 3, respectivamente.

Un estudio detallado de los resultados que nos ofrecen las tablas anteriores, pueden llevarnos a las siguientes consideraciones:

- (1) La traslación pura de un objeto en cualquiera de los tres ejes (x, y, z) es directamente detectable, como muestran los resultados de las imágenes de la tabla 2.
- (2) La rotación de un objeto es interpretable a partir de un cambio suave tanto en los segmentos de línea máximos como en las coordenadas del punto representativo del objeto (ver tabla 1).
- (3) Un movimiento más complejo, como el ofrecido en la secuencia 3 (acercamiento más giro), no es directamente interpretable, aunque sí sea posible imaginar al menos parte del mismo a partir de los parámetros fundamentales que hemos recabado, a saber el tamaño y la localización del objeto en todo instante.

CONCLUSIONES

Se ha presentado en este artículo un método sencillo, pero eficaz, para la detección de dos parámetros importantes de un objeto en movimiento, como son el tamaño y la localización del mismo. El algoritmo es implementable en hardware, utilizando mecanismos neuronales, persiguiendo el objeto de obtención de los datos buscados en tiempo real.

REFERENCIAS

- [Fer92] M.A. Fernandez & J. Mira. (1993). Permanence memory: A system for real time motion analysis in image sequences. MVA'92 IAPR Workshop on Machine Vision Applications, 249-252. Tokyo.
- [Fer93] M.A. Fernandez. (1993). Análisis de movimiento en secuencias de imagen. In: Notas de Visión y Apuntes sobre la Ingeniería del Software, III Curso de Verano de Informática, Colección Estudios, 24, Universidad de Castilla-La Mancha, 99-110.
- [Fer95] M.A. Fernandez, J. Mira, M.T. López, J.R. Alvarez, A. Manjarrés & S. Barro. (1995). Local accumulation of persistent activity at synaptic level: Application to motion analysis. From Natural to Artificial Neural Computation, IWANN'95, 137-143. Springer-Verlag, Germany.
- [Fer97a] M.A. Fernandez. (1997). Una arquitectura modular de inspiración biológica con capacidad de aprendizaje para el análisis de movimiento en secuencias de imagen en tiempo real. Tesis Doctorales, 48. Universidad de Castilla-La Mancha. Ph. D. thesis.
- [Fer97b] A. Fernandez Caballero, M.D. Lozano & A. Pons. (1998). La inspiración biológica de los modelos computacionales de análisis de movimiento de imágenes. Ensayos, 12, Diciembre 1997, Universidad de Castilla-La Mancha.
- [Mit96] A. Mitiche & P. Bouthemy. (1996). Computation and analysis of image motion: a synopsis of current problems and methods. International Journal of Computer Vision, vol. 19, issue 1, 29-55.
- [Mut86a] K. Mutch. (1986). Determining object translation information using stereoscopic motion. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 8, no. 6, 750-755.
- [Mut86b] K. Mutch & L.C. Heiny. (1986). Calculating object size from stereoscopic motion. Proceedings, CVPR86. IEEE Publ. 86CH2290-5, 183-187.