

Sistema de Visión Computacional para Seguimiento de Personas

Saavedra Alcoba Marcelo
Carrera de Ingeniería de Sistemas, Escuela Militar de Ingeniera
La Paz, Bolivia
msaavedra@adm.emi.edu.bo

Computational Vision System for People Tracking

Resumen— El siguiente trabajo se basa en algoritmos de detección de movimiento y seguimiento de objetos para aplicaciones de seguridad mediante los métodos de diferencia de *frames* y de estadísticas simple, mostrando la superioridad del segundo método para aplicaciones de seguimiento de personas, estas operaciones son mejoradas con operaciones morfológicas de procesamiento de imagen, luego se realiza un método simple de seguimiento (tracking) para poder predecir el próximo paso donde se dirige la persona. Este trabajo está realizado con el fin de aplicaciones a la seguridad, donde se obtuvieron buenos resultados.

Palabras Claves— Convolución, filtrado, morfología, detección de movimiento, visión artificial, seguimiento de objetos, varianza, desviación estándar.

Abstract - The following work is based on algorithms of motion detection and object tracking for security applications by means of frame difference methods and simple statistics, showing the superiority of the second method for people tracking applications, these operations are improved with morphological operations of image processing, then a simple tracking method is performed to be able to predict the next step where the person is going. This work is done with the

purpose of security applications, where good results were obtained.

Keywords - Convolution, filtering, morphology, motion detection, artificial vision, object tracking, variance, standard deviation.

I. INTRODUCCIÓN

En el presente artículo se desea realizar un sistema para aplicaciones de sistemas de seguridad por medio de la visión artificial. El utilizar aplicaciones de visión artificial para la seguridad ya es toda una realidad. Por ejemplo, el departamento de defensa de EE.UU. realiza estudios de reconocimiento facial y detección de patrones en base a técnicas de visión artificial buscando la forma más precisa y rentable para reconocer a las personas que puedan suponer un riesgo para la seguridad de dicha nación.

En este artículo se realiza dos métodos de detección de objetos con movimiento. Este tipo de sistemas en su mayoría son utilizados para detectar presencia de personas en lugares de acceso restringido y también para la detección de objetos abandonados o sospechosos en lugares muy transitados. En los últimos años este tipo de sistemas han sido estudiados masivamente en países donde se

tiene un gran temor a atentados terroristas de gravedad.

Este tipo de algoritmos de detección y seguimiento han sido estudiados masivamente y se han desarrollado algoritmos muy eficientes. Entre los algoritmos más conocidos se destacan: *Frame Differencing* [5] (diferencia de frames), *Background Averaging* [1] [6] (Promedio Estadístico de fondo), *CamShift* [3] y *MeanShift* [4] [7].

En la presente investigación se realizan los métodos de *Frame Differencing* y *Background Averaging*, este sistema pretende ser de fácil instalación en una cámara de bajo costo como una webcam. Además de seguir un determinado objeto en movimiento en la escena, independientemente de la posición que ocupe en el entorno, este sistema es capaz de realizar un seguimiento y una predicción hacia donde se dirige el objeto detectado (en este caso personas).

II. MATERIALES Y MÉTODOS

a) Análisis del problema a resolver

El problema a resolver se toma como un ejemplo a un pasillo hacia algún lugar restringido, tomando este caso pueden existir tres diferentes movimientos:

- Se mueve el objeto
- Se mueve la cámara
- Se mueven ambos (cámara y objeto)

Si una cámara se sitúa sobre una superficie alta como el techo interno de un ambiente se puede tener una imagen obteniendo imágenes vistas desde arriba. Dependiendo la cámara que se utilice, se adecua el rango de detección para que se obtenga una imagen completa del ambiente ya que se desea captar todos los posibles movimientos existentes. En este caso se propone mantener la cámara de manera estática y el movimiento será realizado por personas que pasan por debajo del área de detección de la cámara. Para este caso, se utilizará

una base de datos (BD) de imágenes, donde se contiene una serie de imágenes tomadas desde una cámara instalada desde una superficie superior apuntando hacia abajo, esta BD fue creada en la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile.

b) Comparación de los métodos *Frame Differencing* y *Background Averaging*

Como la mayoría de las aplicaciones utilizan una cámara fija, el resultado de la imagen es un fondo estático y el método de movimiento más intuitivo y rápido de detección es el de comparar los cambios de color de píxeles entre dos imágenes (en diferentes tiempos).

Una diferencia de intensidad es generalmente calculada entre dos fotogramas sucesivos o entre un cuadro y una imagen de referencia que no contiene objetos en movimiento [1].

Este detector es el método más simple que existe para la detección de movimiento, el algoritmo a seguir se escribe a continuación:

$$R(x, y) = \begin{cases} 1 & \text{if } I_t(x, y) - I_{t-1}(x, y) > 10 \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Donde $I_t(x, y)$ es el pixel correspondiente a una coordenada xy en el tiempo t , y $I_{t-1}(x, y)$ es el pixel en un tiempo anterior ($t-1$). Se escogió un umbral de 10, ya que se desea tener una detección muy sensible, sin duda esto provoca ruido, pero se obtiene un buen rendimiento para solucionar el problema del ruido se decidió utilizar métodos de morfología matemática.

La morfología matemática es una herramienta muy utilizada en el procesamiento de imágenes. Las operaciones morfológicas pueden simplificar los datos de una imagen, preservar las características esenciales y eliminar aspectos irrelevantes. La idea básica en la morfología es probar una imagen con otra imagen con una forma predefinida simple

obteniendo algunas alteraciones sobre la imagen original. Esta imagen de forma predefinida simple se llama Elemento Estructurante (E.E.), y es en sí misma una imagen binaria (es decir, un subconjunto del espacio o de la cuadrícula). Los elementos estructurantes pueden tener diferentes formas (circular, cuadrada, cruz, disco). Una vez seleccionado el elemento estructurante se realiza una Convolución con la imagen.

Una convolución es un operador matemático que transforma dos funciones “ f ” y “ g ” en una tercera función que en cierto sentido representa la magnitud en la que se superponen f y una versión trasladada e invertida de g . Matemáticamente se define como la integral del producto de ambas funciones después de desplazar una de ellas una distancia η (ver figura 1) [2].

$$f(t) * g(t) = \int_{-\infty}^{\infty} f(\eta) g(t - \eta) \quad (1)$$

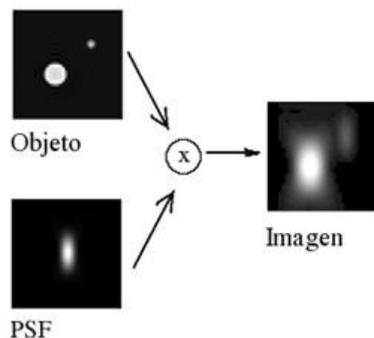


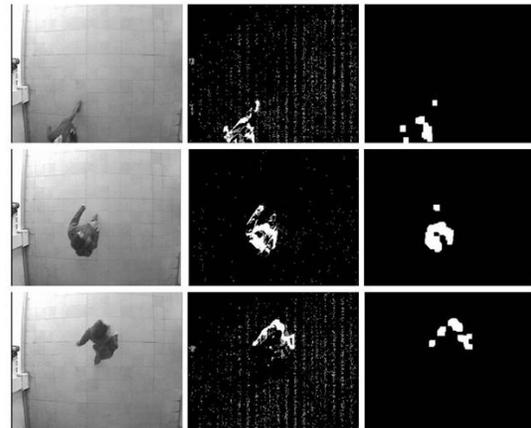
Fig. 1. Convolución de una imagen (donde PSF es un E.E.)

Los operadores básicos en la morfología matemática son: Erosión, Dilatación, Apertura y Cierre.

En el presente trabajo se realiza una combinación de los operadores básicos, y una selección heurística de los elementos estructurantes basados en una observación de los resultados. Los mejores resultados observados, fueron utilizando una operación morfológica de Apertura, seguida de una Dilatación y un Cierre para poder anular el ruido existente y seleccionar mejor al objeto en movimiento.

$$\text{Operador} = (((A \circ B) \oplus B) \bullet B) \quad (2)$$

Donde A = Imagen Original y B un elemento estructurante tipo disco de 5x5



a) b) c)

Fig. 2. Resultados método *Frame Differencing*

a) Imagen original b) resultado sin limpieza c) resultado limpio

En la figura 2 se puede observar en la primera columna, la imagen original, en la segunda columna se muestra los resultados del detector de movimiento y en la tercera columna se observa el resultado de la operación morfológica. Como se ve en los resultados de la tercera columna, la detección es exitosa, pero se tienen varios blobs (manchas) cuando se trata de un solo objeto, esto puede causar problemas ya que los resultados pueden ser confundidos como dos objetos cuando en realidad existe solo uno.

Buscando una mejora al problema anterior, se realizó el método *Background Averaging*, este detector utiliza el promedio y la desviación estándar de cada pixel en un conjunto de datos, en este caso se utilizó las primeras 20 imágenes ya que las primeras 20 imágenes no contienen movimiento y servirán para que el algoritmo pueda aprender el fondo de la escena. Para obtener la desviación estándar se utilizó una aproximación mediante el promedio de las diferencias de las imágenes del fondo, el algoritmo final queda como:

$$R(x,y) = \begin{cases} 1 & \text{If } I_t(x,y) > I_{media_t}(x,y) - 10 * I_{desv_t}(x,y) \\ & \text{and } I_t(x,y) < I_{media_t}(x,y) + 10 * I_{desv_t}(x,y) \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

Se escogió un parámetro de multiplicación de 10 para la desviación estándar y se pudo observar buenos resultados, para eliminar el ruido presente se utilizó la misma operación morfológica de la ecuación (3). Los resultados se despliegan a continuación.

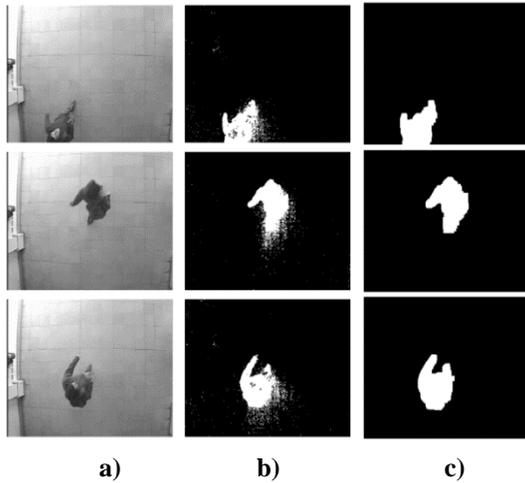


Fig. 3. Resultados método Background Averaging

a) Imagen original b) resultado sin limpieza c) resultado limpio

Como se puede observar en la figura 3, el resultado del método y la operación morfológica en la tercera columna elimina el problema del anterior método, creando un solo blob para una persona.

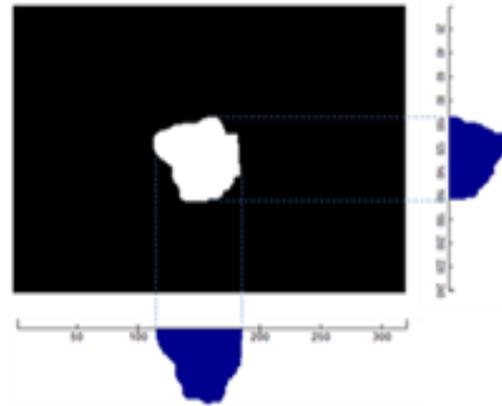
c) Seguimiento del objeto (tracking)

Una vez detectado el objeto, se procede a realizar un algoritmo de seguimiento que pueda indicarnos como se mueve el objeto.

Fig. 4. Histogramas de la imagen en ambas direcciones

El seguimiento consiste en la búsqueda de un punto para detectar su posición y velocidad, con estos resultados se puede estimar el

siguiente movimiento del objeto. Para poder



enmarcar el objeto se necesita 4 puntos que se obtienen mediante dos histogramas del resultado de la ecuación (4).

Mediante los histogramas que se ven en la figura 4, se consigue trazar el área rectangular del objeto. Graficando este rectángulo sobre la imagen original se obtiene el enmarcado del objeto con movimiento (ver figura 5).

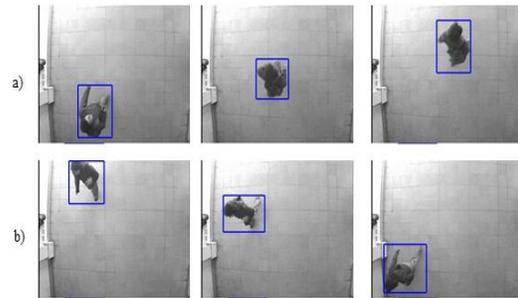


Fig. 5. Enmarcado del objeto con movimiento.

Para la elaboración del algoritmo de seguimiento se realizará un estimador simple prediciendo así la próxima posición del movimiento. El estimador ejecuta restando la posición actual menos la anterior tanto para el movimiento vertical como para el movimiento horizontal, por lo que se tiene un estimador simple de 8 posibles direcciones. Las condiciones para realizar el estimador simple se despliegan a continuación:

Ecuación para la dirección Horizontal:

$$DirH = \begin{cases} 1 & \text{if } PosH_t - PosH_{t-1} > 0 \\ -1 & \text{if } PosH_t - PosH_{t-1} < 0 \\ 0 & \text{if } PosH_t - PosH_{t-1} == 0 \end{cases} \quad (5)$$

Ecuación para la dirección Vertical:

$$DirV = \begin{cases} 1 & \text{if } PosV_t - PosV_{t-1} > 0 \\ -1 & \text{if } PosV_t - PosV_{t-1} < 0 \\ 0 & \text{if } PosV_t - PosV_{t-1} == 0 \end{cases} \quad (6)$$

De la combinación de DirH y DirV se obtienen las 8 posibles direcciones que se muestran a continuación:

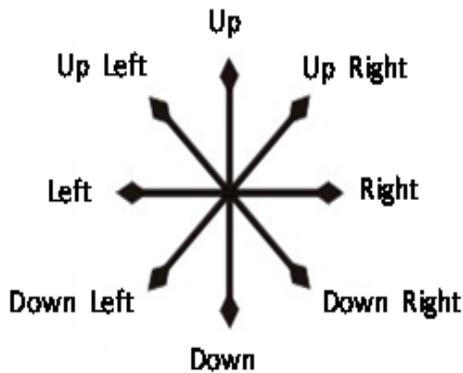


Fig. 6. Ocho posibles direcciones del estimador simple

Cabe destacar que el algoritmo estimador toma como punto medio el centro del rectángulo, es decir el centro masa del objeto, esto no precisamente es el punto central de la persona en movimiento ya que la persona genera sombras y el rectángulo puede crecer en el sentido que se presenta la sombra, para solucionar esto se podría tomar un algoritmo de reconocimiento de texturas, y así eliminar las sombras en base a su textura.

III. RESULTADOS

El resultado del estimador simple se puede ver en la figura 7.



Fig. 7. Estimación de la próxima posición del objeto

Como se puede observar en el gráfico anterior, los círculos de color rojo representan el próximo lugar donde podrá estar el centro del blob. Se puede notar que este algoritmo de estimación es muy sensible a los movimientos bruscos o muy lentos y el movimiento de una persona al caminar puede ser un poco caótico, esto hace que sea un trabajo complejo para un predictor simple, esto se podría mejorar con un estimador más robusto.

A continuación, se despliega casos extremos del Seguimiento mediante el estimador simple:

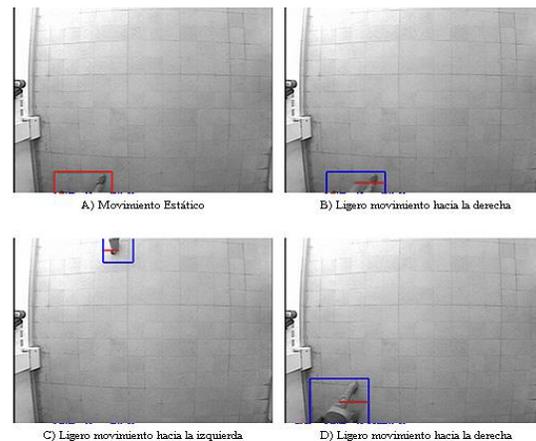


Fig. 8. Estimación de la próxima posición del objeto

En la figura 8 se muestran los datos donde los algoritmos de estimación simple obtuvieron resultados confusos. En A) se muestra un movimiento estático, donde la resta de la posición anterior y la posición actual da cero, en B) y en D) existe un movimiento solo hacia

la derecha, entonces el estimador predecirá que el movimiento es solo hacia la derecha y en el caso de C) la predicción del movimiento apunta a la izquierda de la imagen.

IV. CONCLUSIONES

En la parte de la mejora de la imagen con las operaciones morfológicas, se puede observar que se pueden realizar distintas operaciones para mejorar el ruido existente en las imágenes, tanto como el tipo de operaciones como la forma y tamaño del elemento estructurante.

En la parte de los detectores de movimiento se puede observar que la resta de *frames* es sencillo de implementar, pero el resultado entrega varios blobs, lo que puede traer complicaciones al momento de enmarcar el cuadro o estimar una posición, el método del modelo basado en estadísticas simples es mucho más eficaz ya que entrega un solo blob de la persona en movimiento.

En la parte del Seguimiento se puede observar que el estimador simple funciona correctamente pero solo en 8 posibles direcciones, para mejorar esto y poder tomar más direcciones, sería ideal implementar un estimador mucho más robusto como ser el Filtro Kalman.

REFERENCIAS

- [1] OpenCV 2.1 C++ Reference [online], URL: <http://opencv.willowgarage.com/documentation/cpp/index.html>
- [2] Vignolo Barchiesi Juan, *Procesamiento Digital de Señales*, Ediciones Universitarias Valparaíso, 2008.
- [3] Bradski, G.R., (1998) "Computer Vision FaceTracking for Use in a Perceptual User Interface", Intel Technology Journal, vol. 2, n°2, pp. 13-27.
- [4] Comaniciu, D. y P. Meer, (2002) "Mean Shift: A Robust Approach Toward Feature Space Analysis", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, n°5, pp. 603-619..
- [5] D. Zhang and G. Lu, "Segmentation of moving objects in image sequence: A review," Circuits, Systems and Signal Process., vol. 20, no. 2, pp. 143-183, 2001.
- [6] Michael Oren, Constantine Papageorgiou, Pawan Sinha, Edgar Osuna, and Tomaso Poggio. Pedestrian detection using wavelet templates. In Proceedings of the Conference on Computer Vision

- and Pattern Recognition, pages 193-199, San Juan, Puerto Rico, June 17-19 1997.
- [7] J. A. Corrales, P. Gil, F. A. Candelas y F. "Torres, Diseño de una mini-cámara motorizada para el seguimiento de objetos". Grupo de automática, Robótica y Visión Artificial, Departamento de Física, Ingeniería de Sistemas y Teoría de la Señal, Universidad de Alicante, 2009.

RECEPCION: 12/02/2019
APROBACION: 15/03/2019