

# Segmentación Detección de movimiento

Dan Williams Robledo Cruz  
Visión por Computadora  
Laboratorio de Tecnologías de Informacion,  
Cinvestav  
Cd. Victoria, Tamaulipas  
Email:drobledo@tamps.cinvestav.mx

## I. INTRODUCCIÓN

El objetivo de la practica consiste en segmentar un objeto de interés en una secuencias de imágenes, implementando el método de segmentación por movimiento usando diferencias de imágenes la cual consiste en comparar dos imágenes en un determinado tiempo pixel a pixel.

05 Marzo, 2014

## Implementación

La implementación de de este método fue mediante la realización de un script en MatLab, el cual cuenta con una interfaz gráfica **Fig 1**.

## II. DESARROLLO

### Segmentación del movimiento

La segmentación del movimiento tiene como objetivo la detección de regiones que correspondan a objetos móviles en secuencias de imágenes como por ejemplo vehículos, animales o personas.

Una secuencia de imágenes se define como el conjunto de imágenes de una determinada escena, las cuales son diferentes entre sí, estas diferencias pueden ser interpretadas como variaciones espacio-temporales generadas por el movimiento de los objetos en la escena.

### Diferencia de imágenes

La diferencia de imágenes es uno de los métodos mas sencillos para la segmentación basada en detección de movimiento, hace uso de las diferencias de los niveles de intensidad de los pixeles entre dos tramas consecutivas en una secuencia para extraer las regiones en movimiento y eliminando los componentes estáticos.

Se define como:

$$d_{ij}(x, y) = \begin{cases} 1 & \text{si } |f(x, y, t_i) - f(x, y, t_j)| > T \\ 0 & \text{en otro caso} \end{cases} \quad \text{Donde:} \quad (1)$$

donde  $T$  es un umbral predeterminado.

Los píxeles resultantes en  $d_{ij}(x, y)$  con valor 1 son considerados como resultado del objeto en movimiento, aunque a menudo son causada por el ruido o la iluminación heterogénea.

Normalmente las entradas ruidosas son puntos aislados de la imagen, un método sencillo para eliminarlas consiste en eliminar aquellas regiones que tengan un área menor a un valor predeterminado.

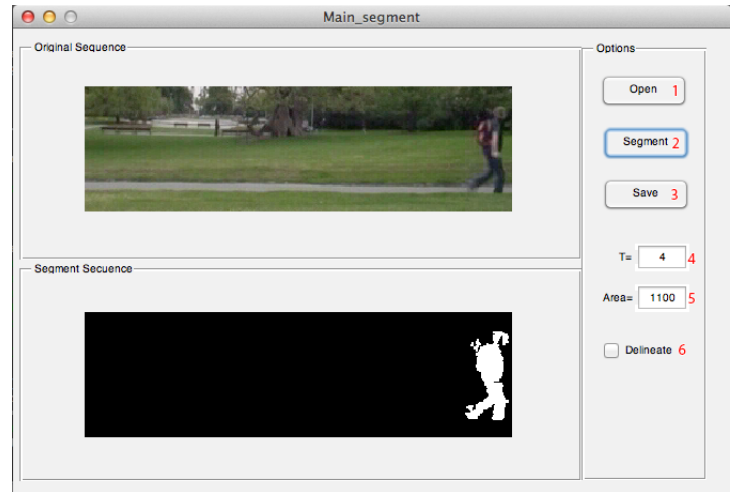


Fig. 1. Interfaz del programa

- 1) Abre una secuencia de imágenes.
- 2) Inicia la segmentación del objeto en movimiento.
- 3) Guarda la secuencia de la imagen segmentada.
- 4) Valor del umbral  $T$ .
- 5) Área de corte, píxeles con un área menor a este valor son eliminados.
- 6) Delinear elemento segmentado.

### Algoritmo 1 Segmentación detección de movimiento

**Entrada:** Secuencia de imágenes,  $T$ ,  $Area$

**Salida:** Secuencia segmentada de imágenes.

```
1:  $Frame = \text{CargarImágenes}()$ ;  
2:  $fondo = Frame(1)$ .  
3:  $SE = \text{strel}('square', 2)$ ;  
4: para  $i = 2$  hasta  $numFrames$  hacer  
5:    $actual = \text{imread}(Frame(i))$ ;  
6:    $dife = \text{abs}(actual - fondo)$ ;  
7:    $I_s(dife > T) = 255$ ;  
8:    $I_s = \text{imopen}(I_s, SE)$ ;  
9:    $I_s = \text{bwareaopen}(I_s, Area)$ ;  
10:   $I_s = \text{imfill}(I_s, 'holes')$ ;  
11: fin para
```

El proceso inicia cargando la secuencia de imágenes al programa, se asigna un valor de nivel de umbral  $T$  así como también un  $Area$ .

A continuación se guarda el primer frame como el fondo estático  $fondo$ , se crea un elemento estructurante  $SE$  para eliminar pequeñas.

Inicio el ciclo principal donde se van leyendo las secuencias de la imagen y se le resta al fondo, si la resta supera el umbral  $T$  se le asigna el valor 255, se le aplica una apertura para eliminar pequeñas regiones de ruido y se rellenan los huecos.

El ciclo se repite hasta acabar con todos los frames.

### Algoritmo 1

### Salida del programa

Los resultados arrojados por el programa se pueden apreciar a continuación.

### Secuencia 1

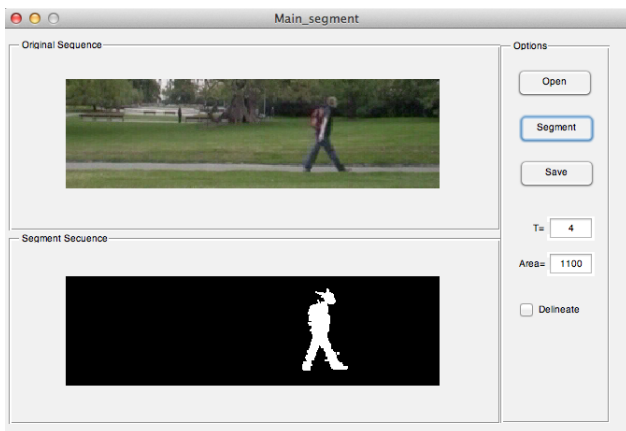


Fig. 2. Secuencia 1 sin delineado,  $T=4$  y  $Area=1100$

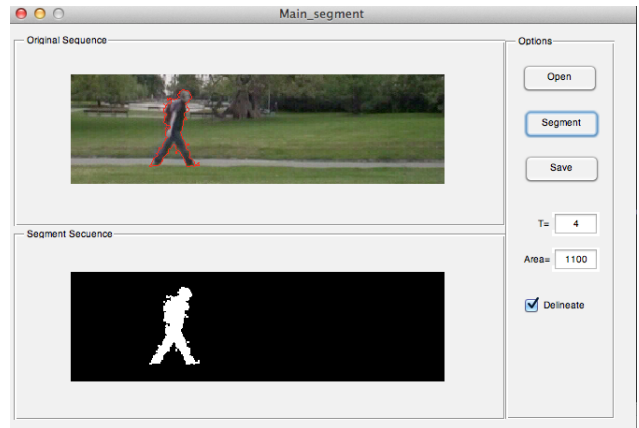


Fig. 3. Secuencia 1 con delineado,  $T=4$  y  $Area=1100$

### Secuencia 2

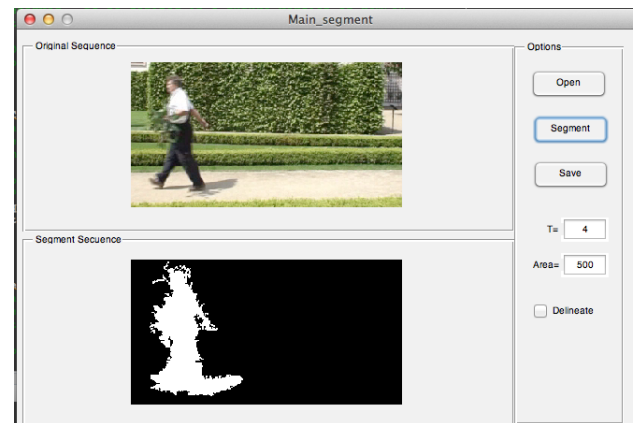


Fig. 4. Secuencia 2 sin delineado,  $T=4$  y  $Area=500$

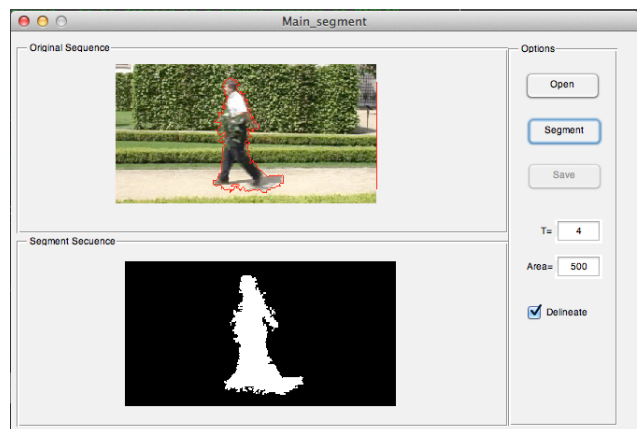


Fig. 5. Secuencia 2 con delineado,  $T=4$  y  $Area=500$