

UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR



MASTER'S PROGRAM IN RESEARCH AND INNOVATION IN INFORMATION
AND COMMUNICATION TECHNOLOGIES (I2-ICT)
Video Analysis Techniques for Surveillance (Course 2012/2013)

LAB 3: Blob extraction and classification

Manual de usuario

Augusto Bourgeat Terán
José Luis Carrillo Medina

Introducción

Este manual permite ejecutar un módulo de análisis de Blob, utilizando segmentación de primer plano que genera una máscara binaria (con valores 0 y 255) y la secuencias de vídeo de entrada capturada por las cámaras de vídeo fijas, y a través de funcionalidades básicas de Blob, obtener la extracción de blob, clasificación de blob y detección de objetos estacionario.

Características de la Implementación:

El módulo implementa lo siguientes

- Extracción de blob,
- Clasificación de blob, y
- Detección de objetos estacionarios.

Las rutinas desarrolladas se integrarán en un solo programa que se implementa en C utilizando la biblioteca pública opencv bajo el sistema operativo Linux.

Los archivos necesarios para la ejecución, puesta en marcha y funcionamiento de Extracción de blob, Clasificación de blob, y Detección de objetos estacionarios son: main.c, blobextract.cpp, blobclassify.cpp, blobstationary.cpp y el video pe.mpg.

El fichero main.c: Es el programa principal el cual declara las variables generales que permiten manipular imágenes (frente, fondo, mascara, blob, estacionario, etc) así como también hace el llamado a las funciones por cada una de las imagen de video.

blobextract.h Es un archivo donde se declara las variables para realizar la extracción de Blob. La extracción se lleva a cabo basándose en el análisis de los componentes conectados A 8 utilizando la función propia de Open Cv (cvFloodFill).

blobclassify.h: Es un archivo donde se declara las variables para realizar la Clasificación de Blob entre las cinco clases disponibles en 'BasicBlob.h' (persona, grupo, auto, objetos y desconocido). Además dibuja blobs (y sus clases) con diferentes rectángulos de colores.

blobstationary.h Es un archivo donde se declara las variables para realizar la extracción de objetos estacionarios en base a técnicas de contadores. Consiste en verificar que un objeto que se está moviendo, que no es parte del fondo, se detiene en un periodo de tiempo se lo detecta como objeto del frente y se reporta como objeto abandonado. Para este caso se tomo un período de tiempo de 10 seg.

PROCEDIMIENTO

1. Directorio de inicio

Ubicarse en el Directorio donde se encuentran los archivos principales: main.c, blobextract.cpp, blobclassify.cpp, blobstationary.cpp BasicBlob.cpp, BasicBlob.h, blobfuncs.h, BlobList.h BlobList.cpp y verificar la existencia del video que queremos realizar la extracción y clasificación de Blob: pe.mpg.

2. Selección del video

Identificar el nombre del archivo (video) que deseamos realizar la extracción y clasificación de Blob.

"mh.mpg"

En el archivo main.h definimos en la constante INPUT_VIDEO el nombre de la secuencia de imágenes (video) que deseamos realizar la extracción y clasificación de Blob.

```
#define INPUT_VIDEO "pe.mpg"
```

Por ejemplo se ha seleccionado "pe.mpg" deshabilitando los otros videos.

Si cambiamos de video, debemos realizar una compilación para que tome en cuenta el cambio de secuencia de video utilizada.

3. Ejecución de los archivos

Compilación de programas

Para compilar el programa digitamos el comando `make`.

Ejecución del programa

Para ejecutar el programa digitamos el comando `./vats`

4. Corridas

Inmediatamente se presenta en pantalla la secuencia de video

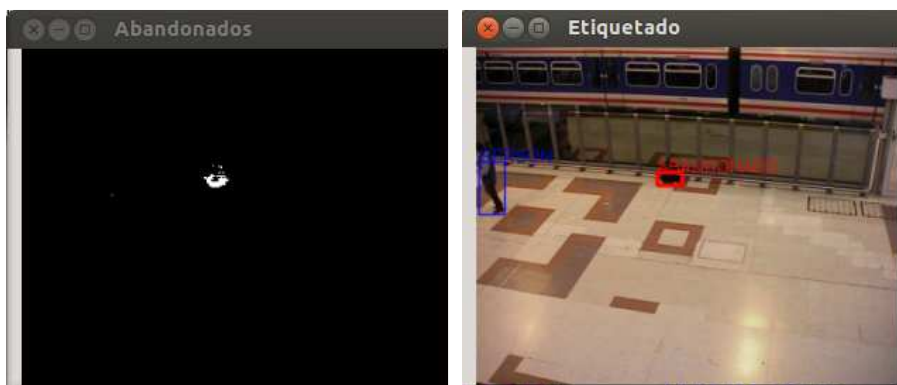
5. Extracción y clasificación de Blob.

Se observará el seguimiento en el resto de imágenes, realizando la Extracción y clasificación de Blob..



Clasificación de personas

6. Detección objeto estacionario



Detección objeto estacionario

Etiquetado

7. Break

Aplastando la tecla `esc` (escape) , en cualquier momento de la ejecución se puede salir del programa.

Caso contrario al finalizar las secuencias del video el programa termina, pudiendo repetir los pasos indicados.

Para el caso de implementaciones adicionales como: compacidad, redes Neuronales existe una carpeta adicional que contiene los mismos archivos y otros como `redes.cpp` `redes.h` (archivos para entrenamiento y clasificación de blob), la ejecución sigue la misma secuencia descrita en el procedimiento, está habilitado el clasificador de redes neuronales.

Estos son los pasos a seguir y los detalles a tener en cuenta a la hora de poner en marcha el programa de extracción de Blob, clasificación de Blob, detección de objetos abandonados y hacer un uso correcto del mismo.

Realizado por:

Augusto Bourgeat
José Luis Carrillo

Program: Master's program in Research and Innovation in Information and Communication Technologies (I2-CIT)

Center: Escuela Politécnica Superior

University: Universidad Autónoma de Madrid
Madrid 2013