

Tarea 2

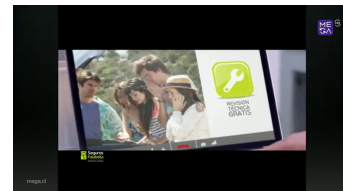
CC5204 – Búsquedas por contenido de Imágenes y Videos 29 de Marzo de 2016

El objetivo de esta tarea es comparar el uso de diferentes descriptores globales para resolver el problema de detección de spots publicitarios (avisos comerciales) en televisión.

Se dispone de varios archivos de video de cuatro horas de duración con la programación emitida por un canal de televisión local. A continuación se muestran algunos cuadros de esos videos:



Por otra parte, contará con un total de 10 cortos de video, cada uno de largo entre 10 y 40 segundos, donde cada video corresponde a un spot publicitario emitido por el canal durante ese periodo. A continuación se muestra un cuadro de ejemplo para tres avisos comerciales diferentes:



Parte 1

Implemente un detector de comerciales, que señale cuantas veces fue emitido cada comercial durante las 4 horas de programación y determine exactamente el inicio y fin de cada aparición. Puede usar el formato de VideoLAN Player (o algún otro formato que le acomode) para mostrar cada emisión detectada:

```
"video.mp4" --start-time=[segundos_inicio] --run-time=[segundos_largo]
```

Para implementar el detector de comerciales deberá decidir un método de descripción del contenido (extracción de características), un método de búsqueda por similitud (comparación de descriptores), y un método de detección (encontrar secuencias parecidas).

Llamaremos una "**configuración**" a la elección de extracción de característica, búsqueda por similitud y método de detección que permiten realizar una detección de comerciales. Una configuración de ejemplo podría ser:

- Etapa de Descripción de Contenido:
 - Extraer tres cuadros por segundo de todos los videos.
 - Dividir cada cuadro en cuatro zonas y calcular histograma de grises normalizado de 128 bins por cada zona.
- Etapa de Búsqueda por Similitud:
 - Comparar histogramas con distancia euclidiana.
 - Para cada cuadro de video de programación buscar el más parecido entre los cuadros de los videos de comerciales (búsqueda k-NN con $k=1$).
- Etapa de Detección:
 - Buscar la secuencia de vecinos más cercanos consecutivos que pertenecen a un mismo comercial.
 - Reportar una detección cuando la secuencia de vecinos más cercanos abarque el 90% de un comercial.

Otra configuración podría ser similar anterior, pero utilizando el descriptor *Ordinal Measurement* de 3x3 zonas y comparando con distancia de Hamming.

En el curso se han estudiado (hasta el momento) diferentes métodos para representar la información visual. Puede implementar cualquiera de ellos en su detector o puede diseñar uno que estime conveniente.

Defina una configuración a su medida, luego impleméntela y reporte todas las emisiones de cada comercial para las 4 horas de programación. Su resultado debe tener una alta efectividad de detección sin falsas alarmas, es decir, **debe reportar la mayor cantidad de comerciales emitidos SIN REPORTAR NINGÚN ERROR**. Una detección es correcta cuando acierta en los tiempos de inicio y fin de cada emisión. No es necesario considerar temas de eficiencia.

Señale la empresa que realizó un mayor gasto en publicidad durante esas cuatro horas de programación (asumiendo valor constante de cada segundo al aire).

Parte 2

Definiremos efectividad como "el número de detecciones correctas sin falsas alarmas", y definiremos eficiencia como "la cantidad de recursos computacionales requeridos para generar el reporte de detecciones", donde los recursos computacionales pueden ser espacio (en disco/memoria) y tiempo de ejecución.

En la parte anterior, cada configuración requiere de diferentes parámetros que afectan su efectividad y eficiencia. Por ejemplo, el número de bins en un histograma o el número de zonas del descriptor Ordinal Measurement.

Compare la efectividad y eficiencia de diferentes configuraciones a su elección y encuentre una configuración óptima que logre una alta efectividad y la mejor eficiencia. Por ejemplo, puede probar con diferentes descriptores, diferentes funciones de distancia, distinta cantidad de cuadros a extraer de cada video, diferentes funciones de distancia, etc.

Concluya qué parámetros son los que más afectan la efectividad y eficiencia en el problema de detección de comerciales. Específicamente, analice el impacto de la dimensionalidad del descriptor y la cantidad de cuadros extraídos de cada video sobre la efectividad y eficiencia. **Justifique sus argumentos mediante gráficos.**

Se dará un bonus de 1 punto en la próxima tarea o control a los integrantes del grupo que logren las configuraciones de mayor eficiencia y efectividad.

Informe

Debe entregar un informe impreso en papel que contenga lo siguientes puntos:

- **Parte 1:**
 - Descripción detallada de la configuración implementada. Describir las etapas de descripción de contenido, búsqueda por similitud y detección.
 - Reporte de comerciales emitidos, incluyendo ubicación en cada archivo de video.
 - ¿Qué empresa gastó mayor cantidad de dinero en publicidad en televisión?
- **Parte 2:**
 - Descripción de las diferentes configuraciones testeadas.
 - Gráficos mostrando el compromiso entre efectividad y eficiencia de las configuraciones testeadas.

- ¿Como se relaciona la dimensionalidad de los descriptores con su efectividad y eficiencia?
- ¿Como se relaciona la cantidad de cuadros extraídos de cada video con la efectividad y eficiencia del detector?
- ¿Cual es la configuración óptima para la detección de comerciales?

Código Fuente

Debe entregar todos los códigos fuentes utilizados para la creación del informe, esto puede incluir (si corresponde):

- Código fuente, en lenguaje C, C++, Java o Python.
- Planillas de cálculo (openoffice, libreoffice, excel, etc.)
- Scripts de comandos (.sh, .bat, etc.)

Debe entregar un archivo `readme.txt` con instrucciones de compilación y ejecución. Notar que siguiendo los pasos detallados en este archivo se deben poder generar los datos que fueron mostrados en el informe, es decir, proveer una serie de pasos o comandos que imprima en pantalla o escriba en archivos el listado de detecciones encontradas para la parte 1 y los datos de los gráficos para la parte 2.

Puede usar cualquier función provista por OpenCV. Recuerde que las dudas sobre programación las puede hacer en el foro de U-Cursos.

Plazo de Entrega

- Código fuente con Informe en PDF: **Jueves 28 de Abril hasta las 23:59** por U-Cursos.
- Un punto de descuento por día de atraso, máximo tres días de atraso.
- Esta tarea la puede hacer en grupos de **máximo dos personas**.

Encontrará los videos que le corresponden a su grupo en la dirección:

<http://juan.cl/cc5204-2016/>