Desarrollo de una biblioteca de código abierto para la realización de video sinopsis

– ENVIO PARA RPIC ESTUDIANTIL –

José Olivera[1]∗, Eduardo Oliva[1], Nelson Ponzoni[1] (ESTUDIANTES),
C. Martínez[1,3] y E. Albornoz[1,2] (DOCENTES ASESORES)

[1]*Instituto de Investigación en Señales, Sistemas e Inteligencia Computacional, sinc(i)*
Facultad de Ingeniería y Cs. Hídricas, Universidad Nacional del Litoral
CC217, Ciudad Universitaria, Paraje El Pozo (S3000), Santa Fe, Argentina
[2]*Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), Argentina*
[3]*Laboratorio de Cibernética, Facultad de Ingeniería, Universidad Nacional de Entre Ríos*
∗ joseolivera123@gmail.com

Resumen— **En los últimos años, la utilización de sistemas de registro de videos se ha incrementado notablemente. Generalmente, estas grabaciones son utilizadas para tareas de vigilancia y permiten monitorear la actividad en una zona particular, rastrear determinadas situaciones o analizar información de incidentes. Estos sistemas almacenan grandes cantidades de datos y la mayor parte de los registros no poseen información relevante, generando una demanda innecesaria de tiempo y dinero para su almacenamiento y análisis. En el presente trabajo se presenta el desarrollo de una biblioteca de código abierto que permite implementar video sinopsis: un resumen automático del video basado en procesamiento digital de imágenes. Con esta herramienta es posible realizar la detección y segmentación de objetos móviles en una escena, para luego generar el resumen del video. Este contendrá la información que se considere relevante y puede presentar la actividad registrada de forma lineal en el tiempo o fusionada sobre el fondo común. Además, se proveen métodos para analizar los objetos y el tiempo en el que desarrollaron su actividad. De esta manera, la biblioteca permite generar resúmenes de videos de actividad selectiva, y como complemento de los sistemas de registros facilita la tarea de los operadores humanos y su implementación.**

Palabras Clave— **video sinopsis, video resumen, segmentación de imágenes.**

1. INTRODUCCIÓN

Los sistemas de video-vigilancia han tenido, en los últimos años, un crecimiento continuo en popularidad y disponibilidad, que lleva a que se capturen videos las 24 hs en infinidad de cámaras alrededor del mundo. Si bien esto permite tener un registro completo de los eventos sucedidos, se hace cada vez más importante el problema de la gestión eficaz del almacenamiento y el análisis de la información almacenada [1]. Dentro de las soluciones aportadas por el procesamiento de imágenes a este problema, el *video analítico* comprende las tecnologías para análisis de eventos específicos en el video, la búsqueda de información para indexado, la recuperación de segmentos particulares, entre otros. En particular, la *video sinopsis* consiste en generar videos de corta duración que contengan únicamente la información relevante proveniente de videos extensos, a partir de ubicar los objetos de interés y fundir su aparición y movimiento en un mismo espacio-tiempo [2].

Los métodos del estado del arte han ido evolucionando desde métodos simples como eliminar cuadros (en inglés, *frames*) cada cierto tiempo (efecto de "cámara rápida") sin fijarse en la actividad de la escena [3], y métodos adaptativos que eliminan cuadros de manera selectiva en períodos de inactividad [4]. Los métodos más recientes están basados en minimizar una función de energía, en cuya función de costo se ponen en juego restricciones de actividad, duración de la sinopsis, fragmentación de objetos, entre otros [5, 6]. En videos de vigilancia (para seguridad urbana, cámaras instaladas en entradas de edificios, etc.), se tiene un escenario común: la captura de objetos móviles de diversos tamaños sobre un fondo relativamente fijo. Así, una metodología difundida de video sinopsis consiste básicamente en modelar el fondo y luego segmentar los objetos, generando secuencias de movimiento individuales que luego son solapadas para su visualización conjunta [7, 8].

Si bien en la literatura se reportan estas técnicas, no se ha presentado y puesto a disposición –hasta nuestro conocimiento– una herramienta software que implemente estos métodos y pueda ser utilizada no sólo para la comunidad científica, sino también por la comunidad en general interesada en esta tecnología y sus aplicaciones. En este trabajo se presenta una biblioteca que implementa un sistema de video sinopsis basada en la detección y seg-

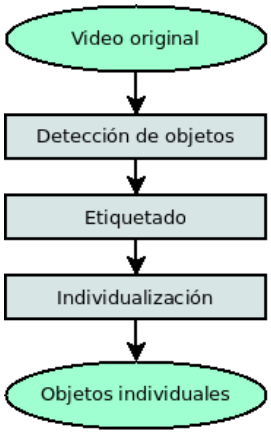

Figura 1: Proceso de detección de objetos en movimiento.

mentación de objetos móviles, su extracción y fusión para generación del video-resumen sobre el fondo modelado.

El resto del trabajo se organiza como se detalla a continuación. En la Sección 2 se resume la técnica de video sinopsis basada en detección de objetos. La Sección 3 presenta los detalles de la biblioteca desarrollada, mientras que la Sección 4 muestra los resultados iniciales y discute algunas de las aplicaciones posibles. Finalmente, la Sección 5 resume los aportes del trabajo y enuncia trabajos futuros.

2. VIDEO SINOPSIS BASADA EN OBJETOS

A continuación se presentan los detalles del método y de la biblioteca desarrollada que permite obtener un resumen de video aplicando técnicas de procesamiento digital de imágenes.

2.1. Modelado del fondo

El objetivo de esta etapa es obtener el fondo estático de la escena, en una imagen que se denomina de "sólo fondo". Esta imagen será utilizada para extraer los objetos "móviles" de la escena y posteriormente, servirá como fondo para superponer los objetos detectados que se deseen, de forma solapada en el tiempo o no. Este fondo se modela, para cada pixel p, mediante una mezcla de Gaussianas $\bar{\eta}_p = (\bar{\mathcal{N}}_p, \bar{\sigma}_p^2)$. A los fines prácticos de este trabajo, se define $\bar{\eta}_p = (\bar{\mathcal{N}}_{i,p}, \sigma_{i,p}^2)$, con $3 \leq i \leq 5$ siendo i el número de gaussianas utilizadas en la mezcla.

$$\bar{\eta}_p = \sum_{j=1}^{i} w_j(\mathcal{N}_{j,p}, \sigma_{j,p}^2) \tag{1}$$

donde w_j representa el peso de la j-ésima componente Gaussiana y $(\mathcal{N}_{i,p}, \sigma_{i,p}^2)$ es la distribución normal de la j-ésima componente dada por

$$(\mathcal{N}_{j,p}, \sigma_{j,p}^2) = \frac{1}{(2\pi)^{\frac{D}{2}} |\sum_j|^{\frac{1}{2}}} e^{-\frac{1}{2}(x-\mu_j)^T \sum_j^{-1}(x-\mu_j)} \tag{2}$$

donde μ_j es la media y $\sum_j = \sigma_j^2 I$ es la covarianza de la j-ésima componente.

Cada cierto intervalo de tiempo T, este fondo es actualizado para evitar que posibles cambios de luminosidad del ambiente u otras condiciones de la escena (por ejemplo, ventanas que se abren o cierran) perjudiquen la etapa de detección de objetos. Esta actualización se realiza de manera recursiva, adaptando el modelo según un parámetro $\alpha \ll 1$:

$$\mathcal{N}_{p,t} = \alpha I_{p,t} + (1-\alpha)\mathcal{N}_{p,t-T}, \tag{3}$$

$$\sigma_{p,t}^2 = \alpha |I_{p,t} - \mathcal{N}_{p,t}|^2 + (1-\alpha)\sigma_{p,t-T}^2. \tag{4}$$

donde $\alpha = 1/T$ es una constante de tiempo que determina el factor de cambio entre frames [9].

Posteriormente, sobre la imagen de fondo se realizan operaciones de suavizado, umbralizado y morfología matemática para lograr un resultado más estable.

2.2. Extracción de objetos en movimiento

Esta etapa puede dividirse en los 3 pasos que están representados en la Figura 1 y que se desarrollan a continuación.

2.2.1. Detección de objetos en movimiento

En este paso se utiliza la imagen de fondo estimada y para cada cuadro I_t (para un tiempo t cualquiera), un pixel es considerado "objeto" si

$$|I_{p,t} - \bar{\mathcal{N}}_p| \geq \epsilon\, \sigma_p^2, \tag{5}$$

para un umbral ϵ dado. Para las imágenes a color, ésto es repetido en cada canal con lo que se generan 3 máscaras binarias que se aplicarán a los canales R, G y B del cuadro I_t. El valor de ϵ otorga al sistema tolerancia al ruido generado en la captura, y es determinado de manera empírica. A partir del modelo de fondo generado, se crea una máscara binaria que se aplica a cada uno de los canales R, G, B de los cuadros del video. Finalmente, se obtiene la envolvente convexa de cada objeto, asegurando una completa inclusión del mismo aún si se perdieron pixeles en el procesado morfológico [10].

2.2.2. Etiquetado de objetos

El objetivo de este paso es que cada objeto identificado en la escena obtenga una única etiqueta, que se mantendrá desde el primero hasta el último cuadro donde aparece el objeto en cuestión. Es decir, que además de etiquetar los objetos, se realizará un seguimiento simple de los objetos en cuadros sucesivos. Para implementar ésto, se computan los centros de masa de los objetos en cada cuadro y se comparan con los calculados en el cuadro previo, mediante la distancia euclídea

$$d_E(C_a, C_b) = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2} \tag{6}$$

donde $C_a = (x_a, y_a)$ y $C_b = (x_b, y_b)$ son los centros de masa de cuadros sucesivos.

Esta distancia es utilizada para chequear la correspondencia de objetos entre frames sucesivos. Si $d_E \leq \delta$, se considera el mismo objeto y la etiqueta asignada es la misma para ambos. Experimentalmente se fija $\delta = 5$. De esta manera, es posible mantener una misma etiqueta para un objeto ya existente y asignar una diferente a los objetos novedosos en la escena [11].

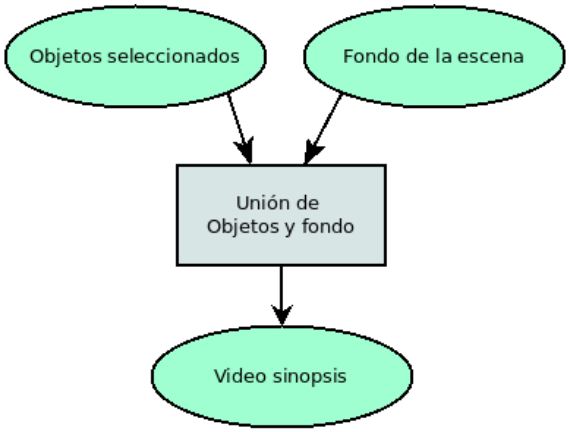

Figura 2: Proceso de fusión de objetos y fondo.

2.2.3. Individualización de objetos

Utilizando el etiquetado previo, se procede a la individualización de cada objeto agrupando aquellos cuadros que los incluyen. De esta forma, para cada objeto se obtiene información acerca de: el instante en que aparece en escena, el tiempo total de permanencia, el momento en que sale de escena, la información espacio-temporal de su trayectoria, su tamaño, su textura, sus colores, etc. Finalmente, para cada objeto se genera un nuevo video donde puede verse el comportamiento de éste (de forma aislada) en la escena. Este conjunto de nuevos videos será utilizado en la siguiente etapa para realizar una fusión con el fondo de la escena.

Es importante destacar que, si bien no se incluye en la versión actual de la biblioteca, puede incluirse un módulo automático de clasificación de objetos. Entonces, en cada uno de estos videos se podría identificar que tipo de objeto está presente en la escena: una persona, un animal, un ciclista, un automóvil, un vehículo de gran porte como ser un colectivo o un camión, entre otros. Incorporar esta categorización sería muy interesante para realizar una fusión condicionada de objetos y fondo.

2.3. Generación de video sinopsis

La sinopsis puede realizarse de dos formas y para ambos casos se consideran: un fondo fijo (en general, uno de los estimados previamente) y la lista de los objetos que se desea que estén presentes en la escena (Fig. 2). La forma más simple de sinopsis genera un video que muestra la actividad en la escena, es decir, que elimina los momentos en que no se detectan movimientos. Con ésto se logra un video compacto que alberga sólo los momentos en que los objetos (semi-automáticamente) seleccionados aparecen en la escena, lo que puede ocasionar la aparición frecuente de superposiciones. Una ventaja de este método es que el usuario no debe establecer ningún parámetro adicional.

La segunda opción de sinopsis permite que los objetos seleccionados estén presentes en escena de manera concurrente. Éstos se irán introduciendo en el mismo orden en que fueron apareciendo en el video original. Además, es posible adicionar un pequeño retraso (k cuadros) en cada nueva inserción para evitar que se superpongan los objetos en el cuadro inicial. La cantidad de objetos simultáneos puede ser controlada para no sobrecargar la escena, mediante un parámetro específico. De esta manera, cuando se alcanza este número prefijado se debe esperar que un objeto salga de escena para incluir uno nuevo. Finalmente, cuando el último objeto sale de escena el video se considera terminado.

Adicionalmente, el sistema dispone de parámetros que controlan el tamaño y color de los objetos que aparecen en escena.

2.4. Biblioteca desarrollada

Todos los métodos necesarios para este desarrollo han sido implementados en C++ utilizando la biblioteca de procesamiento de imágenes OpenCV [12, 13]. Esta última fue escogida por ser una de las más utilizadas actualmente y por poseer una vasta documentación. Las funciones fueron implementadas para ser parametrizables y poder llevar a cabo cada uno de los métodos explicados previamente.

Este desarrollo es de código abierto y está disponible para su descarga en `https://github.com/lerker/videoResumenOpenSource`. Además, se encuentran a disposición algunos videos de ejemplo a partir de los cuales se puede evaluar el desempeño de la biblioteca y con los cuales se pueden reproducir los resultados expuestos a continuación.

3. RESULTADOS

En esta sección se presentan, en primer lugar, algunos resultados obtenidos al utilizar la biblioteca desarrollada. En segundo lugar se discuten algunas aplicaciones que extienden el uso de la misma.

3.1. Experimentos y resultados

Para el desarrollo del trabajo se realizaron pruebas sobre distintos videos, de los cuales se tomó uno como ejemplo para este artículo. Los videos considerados para probar el algoritmo poseen una tasa de 30 cuadros por segundo, con resoluciones de 480p (4:3) y 720p (16:9). Estos fueron tomados tanto de la plataforma para compartir videos YouTube, como también de una base de datos privada de videos de vigilancia. La duración de los mismos fluctuaba entre 10 minutos y 1 hora. En la Fig. 3 se puede observar una escena del video original, donde aparecen entre 3 a 5 objetos (número promedio de objetos concurrentes).

En la video sinopsis generada se puede sobreimprimir una etiqueta numérica que identifica a cada objeto aparece en escena, indicando el tiempo original de aparición. De esta manera se posibilita la identificación de algún objeto que interese, como puede observarse en la Fig. 4.

Finalmente, si se desea obtener algún objeto de tamaño o de un color particular, el método brinda la posibilidad de realizar una segmentación selectiva, obteniendo así

Figura 3: Escena del video de prueba.

Figura 4: Escena de la video sinopsis con objetos etiquetados sobreimpresos.

una video sinopsis con los objetos deseados como puede observarse en la Fig. 5 (ejemplo donde se visualizan sólo objetos de color rojo).

Teniendo en cuenta las pruebas realizadas, las tasas de compresión logradas por el método varían dependiendo de los parámetros seleccionados para ser mostrados en la video sinopsis resultante. En promedio, considerando que los videos utilizados constan de tiempos muertos (sin movimiento), la tasa de compresión lograda fue de al menos 4:1. Ahora bien, modificando parámetros de visualización, como al mostrar objetos de un color determinado, esta tasa aumentó aproximadamente hasta 50:1.

Un video de ejemplo obtenido con la biblioteca desarrollada se puede encontrar online en `https://youtu.be/gKh53lXPlcs` . Se puede observar que el objeto identificado como un automóvil de color rojo aparece en escena a los 7 segundos (Fig. 4), siendo que en el video original, el mismo objeto aparece a los 14 segundos. Por otro lado, se pueden observar ciertos artefactos como oclusiones o superposiciones, producto de los algoritmos empleados para la extracción de objetos en movimiento y fusión con el fondo. En el caso del ejemplo de la Fig. 5, el video puede verse en `https://youtu.be/8CaSpEykyUM`

El método implementado permite eliminar los períodos sin actividad en la escena, logrando una compresión *inteligente* del video original. En consecuencia, sólo se requiere de espacio físico para el almacenamiento de la información relevante.

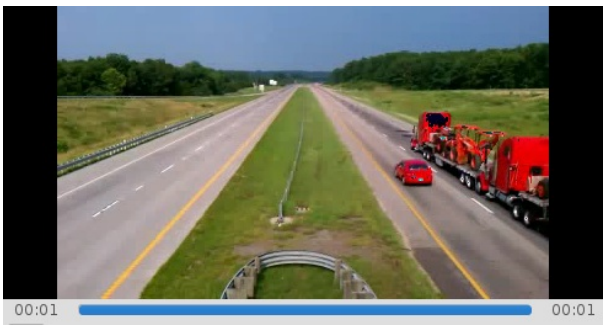

Figura 5: Escena de la video sinopsis mostrando una segmentación de objetos rojos.

4. CONCLUSIONES

En este trabajo se presenta el desarrollo inicial de una biblioteca software de código abierto para la realización de video sinopsis a partir de la detección de objetos. Se proveen implementaciones en C++ para las etapas de todo el sistema: modelado del fondo, detección y etiquetado de objetos, y fusión de movimientos individuales. La biblioteca presenta, además, una gran cantidad de parámetros que se deben configurar para obtener una sinopsis con la información deseada. Estos parámetros están relacionados con condiciones de iluminación, forma de los objetos, color, tamaño, entre otros.

Los resultados preliminares obtenidos son satisfactorios, en tanto se logra eliminar los tiempos muertos (sin actividad) generando un video de menor duración que condensa la actividad detectada. Asimismo, esta biblioteca permitiría abordar y dar solución a problemas de videovigilancia, control de tránsito, seguridad, compresión, etc., en un sistema abierto y de libre disponibilidad.

Como trabajo futuro se plantea desarrollar una interfaz amigable con el usuario, para que el software pueda ser utilizado sin conocimientos específicos de programación o de la estructura interna de la biblioteca. Por otro lado, se pretende agregar funcionalidades de clasificación y agregado de robustez al seguimiento de cada uno de los objetos detectados. Esto permitiría discriminar, por ej., personas y vehículos de diferente porte. Además, se podrían calcular estadísticas individuales (por ej., velocidad de movimiento) o globales (densidad de tráfico, flujo de personas, etc.).

AGRADECIMIENTOS

Los autores desean agradecer a la *Universidad Nacional de Litoral* (con PACT 2011 #58 y CAI+D 2011 #58-511), así también como al *Consejo Nacional de Investigaciones Científicas y Técnicas* (CONICET), de Argentina, por su apoyo.

REFERENCIAS

[1] Thounaojam D. et al. “A survey on video segmentation”. *Intelligent Computing, Networking, and Informatics*. Springer India, 903-912, 2014.

[2] Fu W., Wang J., Gui L., Lu H. and Ma S., "Online video synopsis of structured motion". *Neurocomputing*, vol. 135, pp. 155-162, 2014.

[3] Nam, J. and Tewfik, A., "video abstract of video". In *Multimedia Signal Processing, 1999 IEEE 3rd Workshop on*, IEEE, pp. 117-122, 1999.

[4] Petrovic, N., Jojic, N. and Huang, T., "Adaptive video fast forward". *Multimedia Tools and Applications*, 26(3):327-344, Springer, 2005.

[5] Rav-Acha, A., Pritch, Y. and Peleg, S., "Making a long video short: Dynamic video synopsis". In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 1, pp. 435-441, IEEE, 2006.

[6] Yao, T., Xiao, M., Ma, C., Shen, C. and Li, P., "Object based video synopsis". In *Advanced Research and Technology in Industry Applications (WARTIA), 2014 IEEE Workshop on*, pp. 1138-1141, IEEE, 2014.

[7] Pritch, Y., Rav-Acha A. and Peleg, S., "Nonchronological video Synopsis and Indexing". *IEEE Transactions On Pattern Analysis And Machine Intelligence*, Vol. 30, No. 11, 2008.

[8] Badal, T., Nain, N. and Ahmed, M., "video partitioning by segmenting moving object trajectories". In *Seventh International Conference on Machine Vision (ICMV 2014)*, pp. 94451B-94451B. International Society for Optics and Photonics, 2015.

[9] KaewTraKulPong, P. and Bowden, R., "An improved adaptive background mixture model for real-time tracking with shadow detection". In *Video-based surveillance systems*, Springer US, pp. 135-144, 2002.

[10] Gonzalez, R. and Woods, R., *Digital Image Processing*. New Jersey: Prentice Hall, 2002.

[11] Pritch Y., Ratovitch, S., Hendel A and Peleg, S., "Clustered Synopsis of Surveillance video". *6th IEEE Int. Conf. on Advanced video and Signal Based Surveillance*, Genoa, Italy, 2009.

[12] Laganière, R., *OpenCV 2 Computer Vision Application Programming Cookbook*. Packt Publishing, 2011.

[13] Bradski, G. and Kaehler, A., *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly Media, 2008.