Clustering, en la era de YouTube

Carmen Jara Álvarez
dpto. Ciencias de la Computación e Inteligencia Artificial
Universidad de Sevilla
Sevilla, España
carjaralv@alum.us.es
mcarmenjara@gmail.com

Javier Rodriguez Gallego
dpto. Ciencias de la Computación e Inteligencia Artificial
Universidad de Sevilla
Sevilla, España
javrodgal@alum.us.es
javier21roga@gmail.com

***Resumen*— El primer objetivo de este trabajo, ha sido conseguir un resumen a partir de un vídeo dado, mediante técnicas de clustering, haciendo uso del lenguaje python y de librerías para el tratamiento de imagen. Dicho resumen debe contener entidades y eventos de alta prioridad, exhibir un grado razonable de continuidad, y estar libre de repeticiones.**

El segundo objetivo, pero no menos importante, y que de hecho ha marcado el camino del desarrollo, ha sido comprender el funcionamiento interno del algoritmo utilizado, sus ventajas inconvenientes y posibles soluciones o mejoras.

Palabras Clave*— *Inteligencia Artificial, Algoritmos, K-medias, K-means, Clustering, Summarization, Vídeo, OpenCV, ffmpeg, Machine Learning

I. Introducción

La realización, proviene de la asignación por parte del departamento de Inteligencia Artificial de la universidad de Sevilla, del trabajo de evaluación de la asignatura "Inteligencia Artificial" de tercer curso de Ingeniería del Software.

En la citada propuesta, se pide el desarrollo de un método de resumen de vídeo, en el cual se desarrolle un entorno que permita la lectura y la escritura de fotogramas, y utilizando el algoritmo de K-medias, se calculen los fotogramas clave de una secuencia, así como la configuración de los parámetros del método.[1]

En un primer acercamiento, realizando un estudio previo de algunas de las librerías de Phyton, que la anteriormente mencionada propuesta de trabajo ofrece, se localizó un método optimizado para la realización del algoritmo [2], con el que desarrollamos diversas pruebas satisfactorias.

Sin embargo, tras una consulta con el tutor del trabajo, Don Miguel Ángel Martínez del Amor [3], se decidió abordar el trabajo operando directamente sobre histogramas, de esta forma se conseguía mayor autonomía en el desarrollo, así como una mayor experiencia en el tratamiento imágenes, histogramas y del propio algoritmo utilizado.

Este estudio se enmarca en la problemática actual y de futuro, que supone la rápida implantación de los medios audiovisuales, y más concretamente el vídeo, como elemento de comunicación. Según Cisco, marca emblemática en el ámbito de redes de comunicación, en 2021, el 80% del tráfico de Internet será video. En concreto, prevé la increíble cantidad de tres Zettabytes.[4]

El resumen de vídeos, se hace en este contexto, imprescindible, para aminorar la transferencia de datos.

El abordaje desde k-medias se puede realizar desde múltiples variantes. La estructura de este documento, intenta responder a estas posibles variantes y las decisiones de diseño que se han tomado. En el apartado II (PRELIMINARES), se indicarán las técnicas empleadas y el porqué de ellas, en el apartado III (METODOLOGÍAS) abordaremos el desarrollo del método implementado y posibles variantes, en el apartado IV (RESULTADOS), se analizarán diversos experimentos realizados, desde el punto de vista cualitativo y cuantitativo, extraeremos las conclusiones alcanzadas en el apartado V (CONCLUSIONES) y por último, haremos una propuesta de MEJORA en el apartado VI.

II. Preliminares

Como ya se ha avanzado, en el contexto del trabajo, se van a utilizar las técnicas exigidas por el departamento, en concreto el método de clasificación K-medias.

A. Métodos empleados

Técnicas de planificación: Algoritmo K-medias

Dentro de los numerosos y diferentes acercamientos que se han realizado a la Inteligencia Artificial[5][6][7], uno de los más aplicados en la actualidad es K-medias. Se debe en gran parte a su fácil implementación. Se usa principalmente en minería de datos y como pre-procesamiento para otros algoritmos, por ejemplo para buscar configuraciones iniciales.

K-medias, K-means o clustering es un método de agrupamiento, que tiene como objetivo la partición de un conjunto de observaciones en k grupos en el que cada observación pertenece al grupo cuyo valor medio es el más cercano.

El espacio de datos, queda así dividido en celdas de Voronoi. Planteando aquí nuestro primer gran dilema. El concepto de distancia al hablar de imágenes.

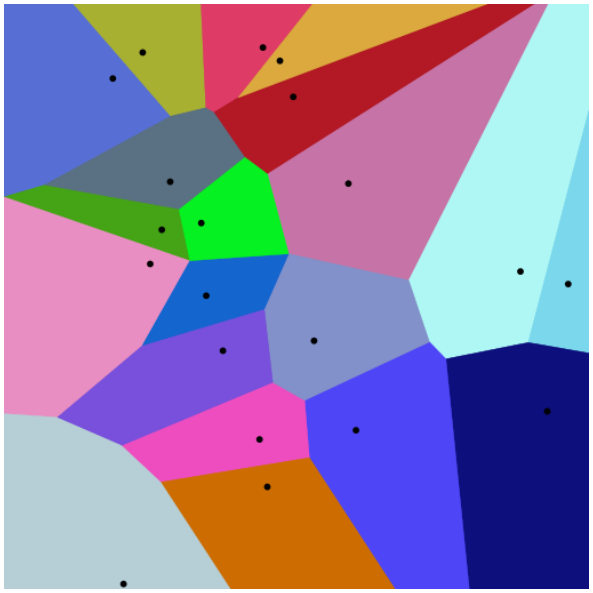

Fig. 1. Regiones de Voronoi en el plano 2D [8]

No será el único problema que se presente, como veremos más adelante.

El algoritmo:

Algoritmo clásico K-medias

Entrada:
- Un número k de clusters, un conjunto de datos $\{xi\}$, y una función de distancia

Salidas:
- U n conjunto de k centros $m1, \ldots, m$ n vector OV con los elementos ordenados

Algoritmo:

1. Inicializar m_i (i=1,...,k) (aleatoriamente o con algún criterio heurístico)
2. REPETIR (hasta que los m_i no cambien):
   2.1 PARA j=1,...,N, HACER:
   Calcular el cluster correspondiente a x_j, escogiendo, de entre todos los m_i, el m_h tal que distancia(x_j,m_h) sea mínima
   2.2 PARA i=1,...,k HACER:
   Asignar a m_i la media aritmética de los datos asignados al cluster i-ésimo
3. Devolver m_1,...,m_n

B. Trabajo Relacionado

Learning problems ans Clusters validations [9]

Finding the number of clusters in a data set: An information theoretic approach. [10]

Data Clustering, 50 Years Beyond K-means [11]

III. METODOLOGÍA

El trabajo como ya se ha comentado se ha realizado en python, implementando un ejecutable, de la siguiente forma:

Resumen de vídeo

Entrada:
- Opción de crear estructura de carpetas en el sistema
- Ubicación de vídeo a procesar
- Ubicación del video procesado
- Número de fotogramas claves (K)
- Tamaño del Histograma (H)
- Saltos en los fotograma para el tratamiento (T)

Salidas:
- Fichero vídeo

Algoritmo:

1. Limpiar resultados anteriores
2. Capturar fotogramas del vídeo
3. Leer fotogramas
4. Obtener histogramas
5. Elegir K centros iniciales (CI)
6. Inicializamos grupos vacíos
7. Para cada fotograma leído (Fi)
   a. Inicializar colección medidas
   b. Tomar histogramas r,g,b
   c. Para cada fotograma inicial(Fj)
      i. Tomar histogramas r,g,b
      ii. Calcular intersección(Fi,Fj)
         1. Calcula intersección roja
         2. Calcula intersección verde
         3. Calcula intersección azul
      iii. Sumar intersecciones
      iv. Almacenar en medidas
   d. Seleccionar máxima medida
   e. Asignar a grupo
8. Inicializamos nuevos centros (NV)
9. Mientras centros iniciales != nuevos centros (CI==CN)
   a. Para cada histograma, r, g,b
      i. Copiar primer histograma
      ii. Acumular histogramas con peso
      iii. Añadir a nuevos centros ficticios
      iv. Buscar fotograma más parecido
      v. Añadir a nuevos centros
   b. Centros iniciales<- nuevos centros
10. Para cada centro
    a. Calcular fotograma más parecido
    b. Copiarlo en carpeta
11. Pasar de fotogramas copiados a vídeo en ruta especificada de salida.

Para iniciar el tratamiento del video, hacemos uso de la librería "ffmpeg"[12].

La mencionada librería, "ffmpeg", nos permite una serie de comandos para que, una vez identificada la ubicación del vídeo de entrada, captemos los fotogramas del mismo y guardemos según ubicación de salida.

No sólo eso, sino que nos permite hacer, previamente a su guardado, un filtrado de los mismos.

De esta forma, y haciendo uso del parámetro de entrada "T", que corresponderá a la determinación del tamaño del salto de selección en los fotogramas originales, guardaremos los seleccionados en una carpeta, previamente creada en el sistema.

Estos fotogramas guardados, serán los que tratemos en el algoritmo, renunciando al resto. El resultado de la utilización de distintos tamaños, lo estudiaremos más adelante.

Para la creación da la carpeta que los contiene y otras necesarias a posteriori, hemos creado una estructura utilizado la librería "os" que nos permite operar desde el entorno de phyton tal como si estuviésemos en la consola del sistema, aportando además herramientas específicas propias [13].

Una vez seleccionados y guardados los fotogramas del video a tratar, procederemos a trasladar su información a histogramas mediante la librería OpenCV [14].

Hemos optado por tratar los histogramas por color pese a que habría otras alternativas como la traducción a grises, que reduciría los datos a tratar y simplificaría la estructura y lectura del programa.

También hemos optado por almacenar los histogramas consecutivamente en una lista continua, de tal forma, que los tres primeros elementos de la lista, corresponden a los histogramas rojo, verde y azul, del primer fotograma, los tres siguientes histogramas a los correspondientes rojo, verde y azul del segundo fotograma, y así sucesivamente.

Los "K" primeros centroides que utilizaremos, los elegiremos de distintas formas: fotogramas consecutivos, fotogramas aleatorios y fotogramas uniformemente distribuidos en el tiempo. Para ello se han realizado tres versiones del ejecutable. Más adelante en la sección de resultados, podremos ver las pruebas realizadas y los mejores obtenidos.

Estamos ya en disposición de iniciar nuestro algoritmo K-medias.

Primeramente, buscaremos nuestra medida de distancia. Para ello, aun existiendo numerosas opciones para comparar histogramas ya implementadas en la librería que estamos usando, OpenCV, y en base a la decisión que hemos tomado de tratar nosotros mismos los histogramas en la medida de lo posible, hemos elaborado un método, que calcula la intersección de dos histogramas.

Para ello, recorremos los ejes de los 2 histogramas de igual tipo (R,G o B), y elegimos el mínimo, que corresponderá al número de coincidencias, como podemos apreciar en la figura (2), que iremos acumulando.

Para la distancia así medida, necesitaremos escoger el máximo como punto más cercano.

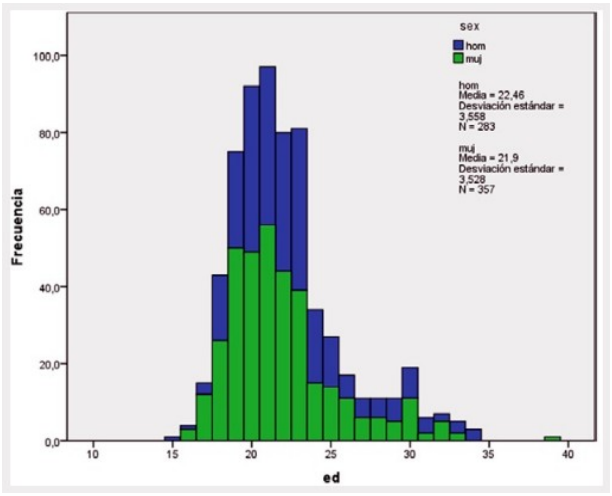

Fig. 2. Histogramas superpuestos [15]

Realizaremos ahora un recorrido de nuestra lista de histogramas general, tomándolos de tres en tres y comparándolos con los tres histogramas de cada centro, color a color.

La suma de las distancias que muestran los histogramas rojos, verde y azul, la guardaremos en una lista, de tal forma que tras acabar las comparaciones con todos los centros, podamos elegir, en nuestro caso el máximo al tratarse de una medida de intersección.

Una vez conformados los grupos, para el cálculo del centro medio, utilizaremos una copia de uno de los histogramas del grupo, sobre el que iremos sumando el resto con peso, nuevamente con la librería "OpenCV". Esto conformará un fotograma medio ficticio, probablemente no existente en nuestro vídeo, que trasladaremos a real, por medio de la función de comparación de intersecciones ya mencionada.

Por último, se inicializa un bucle del que se sale cuando los nuevos centros y los antiguos son coincidentes. En el que los nuevos centros pasan a ser los de referencias y se vuelven a comparar distancias, asignar a grupo y realizar la media.

El traslado a fotograma real que efectuamos en cada iteración, tiene un efecto positivo y otro negativo sobre el resultado esperado. Por una parte, incrementa la posibilidad de caer en mínimos locales, tal como veremos más adelante en el apartado de pruebas, pero por otra parte, nos facilita unos resultados muy aceptables, convergiendo con mayor facilidad.

Una vez que tenemos nuestros fotogramas claves así localizados, realizamos la copia de ellos en diferente carpeta, donde los leeremos y realizaremos el video de salida, nuevamente con herramientas de las librerías "os" y "ffmpeg"

IV. RESULTADOS

Las primeras pruebas las realizaremos para comprobar las distintas distribuciones de los centros iniciales que hemos implementado: de forma aleatoria, selección consecutiva, y uniformemente distribuida en el tiempo. Hemos mantenido fijos el resto de variables, para obtener una mejor detección de los cambios en los resultados.

K utilizada: 15

T utilizada: 6

H utilizada: 256

Vídeos utilizados: V70.mpg y V53.mpg de la base de datos VSUMM dataset [16].

El vídeo V70.mpg contiene grupos de imágenes claramente diferenciables con la siguiente composición:

Tabla 1. Secciones del vídeo V70

Sección	**V70**	
	Descripción	***Duración***
1	Fondo oscuro	0.5 s
2	Espuma blanca	11 s
3	Rocas	6 s
4	Hielo y mar azul	10 s
5	Mar azul	7 s
6	Mar anaranjado	3 s
7	Mar azul	6 s

Los tiempos obtenidos para cada sección son aproximados y pueden variar de un reproductor a otro, pero la proporción entre ellos, que es lo que nos interesa, no varía.

Por lo que, para evaluar según las distintas distribuciones de centros iniciales utilizaremos una evaluación descriptiva de los resultados y el tiempo de ejecución que ha necesitado el algoritmo para terminar y producir el vídeo resumido.

Tabla 2. Evaluación según centros iniciales en V70

Centros	**Resultados V70**	
	Evaluación	***Tiempo***
secuenciales	De los quince fotogramas claves obtenidos, 13 corresponden a fotogramas muy similares de la primera parte del video, quedando por tanto muy descompensada la representación.	2.607 s
distribuidos	Los fotogramas claves obtenidos son claramente representativos del vídeo original, tanto en cuanto al grupo al que corresponde como en la cantidad según los tiempos iniciales	2.645 s
aleatorios	Los fotogramas claves obtenidos tienen una distribución similar a la anterior, si bien la segunda sección toma algo más de peso.	3.141 s

Igualmente, exponemos la composición y resultados de V53.mpg:

Tabla 3. Secuencias del vídeo V53

Sección	**V53**	
	Descripción	***Tiempo***
1	Fondo oscuro con variaciones	18 s
2	Representación corte globo terraqueo	11 s
3	Nieve y árboles	15 s
4	Personas en movimiento	7 s
5	Persona azul	3 s
6	Nieve anocheciendo	7s

Tabla 4. Evaluación según centros iniciales en V53

Centros	**Resultados V53**	
	Evaluación	***Tiempo***
secuenciales	De los quince fotogramas claves obtenidos, 13 corresponden a fotogramas muy similares de la primera parte del video, quedando nuevamente descompensada la representación.	5.080 s
distribuidos	Los fotogramas claves obtenidos son claramente representativos del vídeo original en cuanto al grupo al que corresponde pero los tres últimos grupos están infrarepresentados	6.258 s
aleatorios	Los fotogramas claves obtenidos tienen una peor distribución respecto a los grupos, quedando sin representación las tres últimas secciones..	5.068 s

En general, podemos ver como la elección de centros consecutivos, pese a ser el más eficiente en tiempo de computación, en nuestro programa nos lleva a la localización. La elección del fotograma más parecido, nos arrastra hasta el grupo del que parten los fotogramas iniciales. Desechamos por tanto este método, por ser poco representativo del vídeo.

A simple vista podríamos afirmar que una distribución uniforme sería la más adecuada, y es la que utilizaremos a partir de ahora, ya que es algo mejor en ambos casos en la evaluación y el tiempo, respecto al resultado de centros aleatorios. Pero necesitaríamos ampliar la muestra con vídeos más dispares. Ya que intuimos, que en algunas ocasiones, caso de vídeos cíclicos, podría llevarnos a peores soluciones.

Tras unas primeras pruebas con películas propias durante el desarrollo en la que los resultados variaban del éxito al fracaso a ojos vista de forma aleatoria según el número “K” utilizado, se decidió buscar para las pruebas una película, claramente clasificable de modo natural. Se trata nuevamente de V70.mpg de la base de datos VSUMM dataset [16] con poco movimiento y con una división en varios tiempos claramente diferenciados

en color, que además incorpora una repetición de imágenes, que nos ayudará a la comprobación de calidad del resultado.

Como ya hemos avanzamos, realizaremos las pruebas con el modelo de elección de centros iniciales distribuidos y la compararemos en estos primeros casos con la elección de centros iniciales aleatorios.

Fijamos T y H, variamos K:

T utilizada: 6

H utilizada: 256

Tabla 5. Evaluación según elección de K en V70 con centros distribuidos

K	Resultados V70		
	Evaluación	***Bucles***	***Tiempo***
4	Fotogramas claves pero obvimente no todas las secciones del video representadas.	5	2.993 s
6	Consigue los 6 fotogramas significativos. Faltaría el último	4	3.086 s
7	Aparecen los 6 fotogramas significativos y uno más del grupo de más tiempo. Mar azul. El resultado no se ajusta a la distribución de fotogramas esperada sin embargo, en la distribución en tiempo nos encontramos con el que la secuencia final que es un tema repetido, no se encuentra.	3	3.379 s
15	Correcto. El resultado es un vídeo altamente ajustado al original.	3	2.916 s
30	Correcto. El resultado es un vídeo altamente ajustado al original.	5	5.589 s
100	Correcto. El resultado es un vídeo altamente ajustado al original.	3	7,450 s

Realizamos la misma prueba para centros iniciales aleatorios.

Tabla 6. Evaluación según elección de K en V70 con centros aleatorios

K	Resultados V70		
	Resultado	***bucles***	***Tiempo***
4	4 fotogramas significativos representados.	2	1.879 s
6	Solo 4 secciones representadas.	4	2.920 s
7	Falta la primera sección. El resto crrectas	5	3.596 s
15	Correcto pero aun descompensados los pesos.	5	4.128 s
30	Correcto	7	7.221 s
100	Correcto	5	11.557 s

El porqué de este resultado, hay que buscarlo en uno de los conocidos problemas del método K-medias.

El número de grupos k es un parámetro de entrada fundamental que condiciona y, de hecho, puede llevar a malos resultados. Se espera en K-means, que los grupos tengan igual tamaño, por lo que la asignación al grupo más cercano sería la asignación correcta, sin embargo, nada más lejos de la realidad en el amplio mundo audiovisual. Dudamos que encontremos películas con, por ejemplo, planos de igual duración, tal que pudiésemos agrupar en igual número de fotogramas. Aun existiendo, es una restricción imposible de asumir en el resumen de vídeos.

¿Cómo realizaría la mente humana una división en cuatro grupos cuando a la vista clasifica en tres?

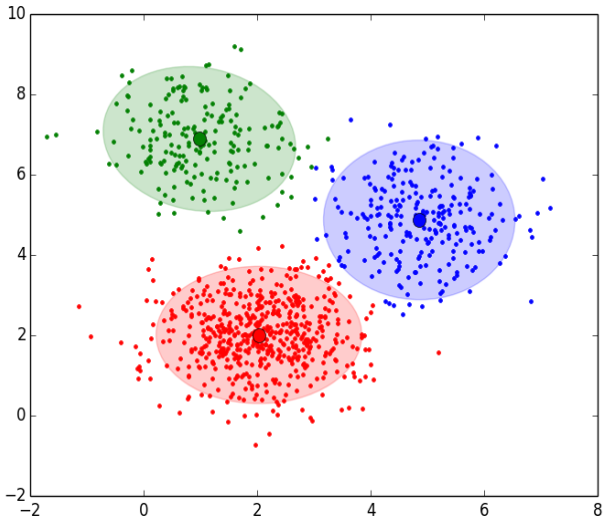

Fig. 3. Distribución idonea para tres clusters [17]

En esta imagen, una clasificación de cuatro, sería claramente poco adecuada. Figura 4.

Existen actualmente webs, que ofrecen herramientas en línea con las que realizar visualizaciones gráficas de resultados con distintas distribuciones de puntos y posibilidad de elección de centros que pueden ayudar a entender el problema [18].

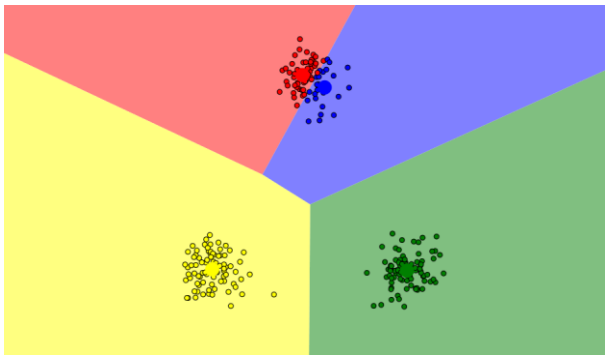

Fig. 4. K 4 para tres clusters

Para la evaluación del resultado según el tamaño del histograma, dejaremos fija nuestra K en un número adecuado, en este caso, según las pruebas iniciales, tomaremos 15 de tal forma que no interfiera.

Centros iniciales distribuidos:
K utilizada: 15
T utilizada: 6

Tabla 7. Evaluación según elección de H en V70 con centros distribuidos

H	Resultados V70		
	Resultado	***Bucles***	***Tiempo***
1	Hemos forzado la parada a las 80 interaciones. El resultado es adecuado.	+80	+ 51.685 s
2	Correcto, si bien la sección 4. Hielo está infrarepresentada	7	5.371 s
5	No queda representada la primera secuencia, el resto distribución correcta.	4	3.452 s
50	Correcto	4	3.520 s
128	Correcto	3	2.932 s
256	Correcto	3	2.916 s
600	Correcto	3	2.981 s
5000	Correcto	3	3.622 s

Realizamos la misma prueba para centros iniciales aleatorios**.**

Tabla 8. Evaluación según elección de H en V70 con centros aleatorios

H	Resultados V70		
	Resultado	***bucles***	***Tiempo***
1	Hemos forzado la parada a las 80 interaciones. El resultado es adecuado.	+80	+51.751 s
2	Las siete secciones. Hielo infrarrepresentado. Espuma suprarepresentada.	6	7.410 s
5	No queda representada la primera secuencia y el peso no es adecuado en el resto.	6	4.667 s
50	No queda representada la primera secuencia y el peso no es adecuado en el resto	7	5.262 s
128	No queda representada la primera secuencia y el peso no es adecuado en el resto.	7	5.319 s
256	Correcto pero aun descompensados los pesos.	5	4.128 s
600	No queda representado la primera y el peso no es adecuado en el resto.	6	5.065 s
5000	Faltan dos secciones, y pesos descompensados totalmente.	5	5.360 s

Ahora fijamos K y H, variamos T:

K utilizada: 15

H utilizada: 256

Tabla 9. Evaluación según elección de T en V70 con centros distribuidos

T	Resultados V70		
	Resultado	***Bucles***	***Tiempo***
1	Falta el fondo oscuro del principio y la sección de la espuma blanca está infrarrepresentada.	8	34.790 s
2	Falta sección inicial aunque en este caso la distribución está compensada.	5	11.601 s
4	Falta sección inicial aunque en este caso la distribución está compensada.	5	5.956 s
6	Correcto. Con todas las secciones y bien distribuidas.	3	2.687 s
20	Representa todas las secciones, aunque deja la del mar azul muy poco representada.	3	1.083 s
40	No representa la última sección, aunque las secciones representadas se encuentran bien distribuidas.	2	0.572 s

Realizamos la misma prueba para centros iniciales aleatorios**.**

Tabla 10. Evaluación según elección de T en V70 con centros aleatorios

T	Resultados V70		
	Resultado	***Bucles***	***Tiempo***
1	Falta una sección. Primeras secciones infrarepresentadas.	6	26.977 s
2	Falta sección inicial. Descompensados los pesos.	4	10.234 s
4	Falta sección inicial. Descompensados los pesos.	4	5.590 s
6	Correcto pero aun descompensados los pesos.	5	4.128 s
20	Correcto pero aun descompensados los pesos.	4	1.338 s
40	Correcto pero aun descompensados los pesos.	3	0.717 s

Para corroborar los resultados obtenidos, realizaremos las mismas pruebas con la elección de centros iniciales distribuida en el video V53.

Fijamos T y H, variamos K:

T utilizada: 6

H utilizada: 256

Tabla 11. Evaluación según elección de H en V53 con centros distribuidos

K	Resultados V53		
	Evaluación	***Bucles***	***Tiempo***
4	Salen 4 fotogramas bien diferenciadas	4	4.203 s
6	Faltan 2 secciones	5	4.719 s
7	Continúa faltando 2 secciones y la distribución no es adeuada.	5	4.764 s
15	Aparecen representadas todas las secciones. Distribución levemente desajustada	6	6.527 s
30	Correcto	4	6.063 s
100	Correcto	4	12.540 s

Ahora fijamos K y T, variamos H:

En este caso, puesto que las pruebas iniciales nos han mostrado que el valor óptimo de K es 30, utilizaremos dicho valor para las siguientes pruebas.

K utilizada: 30

T utilizada: 6

Tabla 12. Evaluación según elección de H en V53 con centros distribuidos

H	Resultados V53		
	Resultado	***bucles***	***Tiempo***
1	Hemos forzado la parada a las 80 iteraciones. Correcto, con una leve descompensación en el peso de las secciones.	+80	+ 94.532 s
2	Correcto	8	10.310 s
5	Correcto	11	13.744 s
50	Correcto	7	9.180 s
128	Correcto	4	5.997 s
256	Correcto	4	6.132 s
600	Correcto	4	6.529 s
5000	Correcto	4	8.814

Para terminar esta prueba, fijamos K y H, variamos T:

K utilizada: 30

H utilizada: 256

Tabla 13. Evaluación según elección de T en V53 con centros Distribuidos

T	Resultados V53		
	Resultado	***Bucles***	***Tiempo***
1	Correcto	5	41.874 s
2	Correcto	5	21.094 s
4	Correcto	4	9.090 s
6	Correcto	4	6.672 s
20	Correcto	4	2.290 s
40	Representativa, pero los pesos de las secciones últimas infrarepresentados.	4	1.323 s

Por último, realizaremos las mismas pruebas en una película en blanco y negro, para ello utilizaremos nuestra ya conocida V70.mpg convertida. Pero obviamente, incluso a simple vista resulta difícil identificar el número de fotogramas significativos mínimos. Todo es más difuso ya que no se diferencian con tanta claridad las distintas secciones que habíamos definido anteriormente.

Fijamos por tanto T y H, variamos K:

T utilizada: 6

H utilizada: 256

Tabla 14. Evaluación según elección de T en V70 Blanco y Negro

K	Resultados V70 blanco y negro		
	Evaluación	***Bucles***	***Tiempo***
4	Representa correctamente 4 de las secciones.	3	2.849 s
6	Representa correctamente 6 de las secciones.	3	2.741 s
7	Aparecen los 6 fotogramas significativos y uno más del grupo de Hielo. El resultado no se ajusta a la distribución de fotogramas esperada, en la distribución en tiempo nos encontramos con el que la secuencia final que es un tema repetido, no se encuentra.	3	2.738 s
15	Falta una sección, por lo demás, la representecaión es adecuada.	3	3.147 s
30	Forzada la parada a las 80 iteraciones. Continua faltando una sección y la distribución es bastante mala.	+80	+77.616 s
100	Correcto	4	9.872 s

Fijamos ahora K y T, variamos H:

Puesto que lo que pretendemos en este caso es comparar los resultados del vídeo V70 en blanco y negro con el V70 a color, utilizaremos la K usada en la prueba anterior, aunque no sea la que ha obtenido el mejor resultado.

K utilizada: 15

T utilizada: 6

Tabla 15. Evaluación según elección de H en V70 Blanco y negro

H	Resultados V70 blanco y negro		
	Resultado	***Bucles***	***Tiempo***
1	Fata sección y pesos descompensados.	+80	56.449 s
2	Fata sección y pesos descompensados.	7	5.741 s
5	Falta sección pero la distribución mejora notablemente.	3	3.143 s
50	Falta sección pero la distribución vuelve a mejorar.	+80	55.690 s
128	Igual resultado que anterior.	3	3.117 s
256	Igual resultado que anterior	3	3.137 s
600	Igual resultado que anterior	3	3.201 s
5000	Igual resultado que anterior	3	3.893 s

Para terminar con la prueba en blanco y negro, fijamos K y H, variamos T:

K utilizada: 15

H utilizada: 256

Tabla 16. Evaluación según elección de T en V70 Blanco y negro

T	Resultados V70 blanco y negro		
	Resultado	***Bucles***	***Tiempo***
1	Fata sección y pesos algo descompensados.	4	21.466
2	Correcto	7	17.297
4	Falta sección y mala distribución.	5	6.610
6	Fata sección..	3	3.136
20	Correcto	3	1.260
40	Correcto	2	0.698

Adicional a las pruebas anteriores, hemos decidido realizar una prueba para un resultado más continuo con el vídeo V53.mpg

Para ello hemos utilizado:

K = 600
T = 2
H = 256

Con estos parámetros obtenemos un vídeo de duración similar al original, pero el cual ocupa aproximadamente la novena parte. Esta prueba se ha realizado con los tres tipos de elecciones de centros iniciales definidos anteriormente:

Tabla 17. Evaluación según centros iniciales en V53

Centros	**Resultados V53**		
	Evaluación	***Bucles***	***Tiempo***
secuenciales	El vídeo queda cláramente descompensado, las primeras secciones cobran una grandísima importancia, durando incluso más tiempo que en el original, mientras que las últimas secciones se reducen significativamente, pasando de 17 s originalmente a 1s	4	166.744 s
distribuidos	Curiosamente, en este caso pasa exáctamente lo mismo que en el secuencial, dejando las últimas secciones sin prácticamente representación	4	179.131 s
aleatorios	A pesar de ser el más lento de los tres, el resultado final del vídeo es realmente muy bueno, mantiene casi en su totalidad la distribución entre secciones, bajando hasta nueve veces el peso del vídeo original.	+80	3317.76 s

V. CONCLUSIONES

Como hemos podido observar, los parámetros incluidos, afectan importantemente al resultado en calidad y eficiencia.

Al ser los vídeos generalmente secuenciales en cuanto a tomas se refiere, proponemos como elemento óptimo, la inicialización de centros distribuidos uniformemente.

En cuanto al parámetro K, número de fotogramas claves, que muestra a medida que se incrementa una relación directa con la calidad de resultados e inversa con el tiempo de computación, para películas pequeñas, similares a las estudiadas, un número de 15, podría cubrir la totalidad de elementos significativos sin deteriorar excesivamente la eficiencia.

Para el parámetro T, con relación directa con la eficiencia y con poca afectación a resultados, proponemos un número alto, por encima de 6. Llevado al extremo, podríamos obviar el propio algoritmo k-medias, y convertirse en sí mismo en el método de resumen.

El parámetro H, tamaño del histograma, incidente principalmente en el tiempo, ha mostrado su mejor comportamiento para el valor 128.

De la propuesta de medida de distancia, nos resulta especialmente difícil en este contexto, y sus posibles óptimos dependerán del tipo de vídeo con el que tratemos.

La medida sumamente simple que hemos realizado de la suma de intersección por colores, no nos ha parecido, ni mucho menos, la mejor, pero en cualquier caso, el método da resultados aceptables.

En general, podemos decir, que aisladamente, sin más ampliación, a pesar de las ventajas de implementación que ofrece, los problemas que conlleva el uso de k-medias están lejos de resolverse en el tratamiento de imágenes, es más, se acrecientan considerablemente por la propia complejidad implícita en el tratamiento de las mismas.

No queremos dejar de señalar algo que durante todo el proceso, nos ha llamado poderosamente la atención que elementos que al ojo humano eran claramente significativos pero breves en tiempo (por ejemplo, en el vídeo V70.mpg hay un corto espacio de tiempo en el que la imagen se oscurece y es atravesada por el sol), en ningún caso se ha determinado como significativa. Obviamente, nos encontramos con el mencionado problema de que K medias supone los grupos de igual tamaño.

Proponemos por tanto, para un siguiente trabajo, otro método que mida el cambio entre un fotograma y el siguiente y destaque así los cambios bruscos de secuencias.

Nos quedamos también con la idea de utilizar como elemento de estudio para un futuro, el incorporar tratamiento de imagen de forma previa. OpenCV nos facilita esta labor mediante: ecualización previa del histograma [19], binarización de imagen con el algoritmo de Thresholding [20],…

Especialmente interesante nos ha parecido la herramienta que nos ofrece OpenCV para la transformada de Fourier [21]

Pero quizás, lo más sugestivo, podría ser explorar nuevos algoritmos, descendientes de K-medias, como son Fuzzy Clustering, donde cada elemento tiene un grado de pertenencia difuso a los grupos o EM algorithm, que es el mismo algoritmo, pero asume distribución gaussiana para los clusters, en lugar de la distribución uniforme.

También podríamos realizar un pre-cálculo del "K" adecuado con el método del codo (Elbow Method), en el que se utilizan los valores de la inercia obtenidos tras aplicar el K-means a diferente número de Clusters, siendo la inercia la suma de las distancias al cuadrado de cada objeto del Cluster a su centroide (1).

$$\text{Inercia}=\sum \| x_i-\mu \|^2 \quad (1)$$

Una vez obtenido los resultados, se podría representar y ver gráficamente donde se produce un cambio brusco de tendencia (codo) [22]. Este pre-cálculo obviamente es computacionalmente costoso, pero podría ser interesante en determinadas condiciones.

Por último, se nos antoja interesante, como elemento de estudio, ver si k-means, como pre-procesamiento de forma previa al uso de otros métodos, más caros computacionalmente, mejora los resultados y eficiencia de estos últimos.

VI. MEJORA

Según las propuestas que se nos ofrece desde el departamento de Inteligencia Artificial, y en base a las necesidades o deficiencias que hemos detectado en nuestros resultados, hemos decidido realizar una heurística que contemple el número de fotogramas, K, de forma automática y que a su vez, intente representar los elementos significativos de corta duración.

Basado en el principio de cierta localidad que ya hemos mencionado, que conlleva la elección de centros iniciales, hemos realizado, un algoritmo de detección de cambios, de tal forma, que detecte saltos de secuencia, mediante diferencia entre imágenes.

Pero por sí solo, no nos serviría para determinar el peso de cada secuencia, es decir, buscamos que el resultado, además incorpore más centroides en los grupos mayores. Para ello, hemos incorporado al algoritmo una fórmula para que evite la necesidad de T y en función del tamaño del video se escoja K.

Este nuevo algoritmo tomará la totalidad de los fotogramas iniciales, y añadirá a los calculados centroides representativos del cambio, uno nuevo, siempre que el grupo que está tratando supere en longitud el 15% del tamaño de los fotogramas iniciales.

Hemos realizado las pruebas sobre el vídeo v70.mpg que contiene 1405 fotogramas iniciales.

Hemos fijado:

H=256

K auto calculada

% es un parámetro y definirá la brusquedad del cambio

% ajuste	**Resultados V70 blanco y negro**		
	Evaluación	***Bucles***	***Tiempo***
50	K=18 Resultado: representativo, con leve suprarepresentación de la secuencia blanca.	13	53.080 s
95%	K=451 Resultado bastante satisfactorio, con buena distribución y representación de todas las secuencias	19	836,154 s

Algoritmo cambio de secuencia

Entrada:
- Lista de histogramas
- Porcentaje de ajuste(%)

Salidas:
- U n conjunto de centroides de longitud variable

Algoritmo:
1. Recorrer histogramas
    a. Calcular distancia histograma i e histograma i+1
    b. Anexar distancia a lista de distancias
2. Calcular media de distancias (media)
3. Recorrer lista de distancias
    a. Si la distancia es menor que el porcentaje de ajuste de la media (%*media) el histograma en curso se anexa a lista de centroides.
4. Incorpora último histograma a la lista de centroides

REFERENCIAS

[1] Página web del curso IA de Ingeniería del Software. https://www.cs.us.es/cursos/iais-2017/trabajos/Clustering.pdf. Consultada el 18 de Junio de 2018.

[2] Página oficial de OpenV. https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_ml/py_kmeans/py_kmeans_opencv/py_kmeans_opencv.html Consultada el 18 de Junio de 2018.

[3] https://www.cs.us.es/~mdelamor/

[4] Página de noticias de tecnología Cisco. Predicciones de tráfico global. https://newsroom.cisco.com/press-release-content?type=webcontent&articleId=1853168 Consultada el 18 de Junio de 2018.

[5] S. Russell, P. Norvig, Artificial Intelligence: A Modern Approach, 3rd ed, Pearson, 2010.

[6] Y. LeCun, Y. Bengio, G. Hinton. "Deep Learning", Nature, vol. 521, 2015, pp. 436-444.

[7] Página web del curso IA de Ingeniería del Software. https://www.cs.us.es/cursos/iais. Consultada el 18 de Junio de 2018.

[8] https://es.wikipedia.org/wiki/Pol%C3%ADgonos_de_Thiessen#/media/File:Euclidean_Voronoi_diagram.svg Consultada el 19 de Junio de 2018.

[9] Pardo, Mateo. Clustering. Nation Research Council (CNR), Berlín http://lectures.molgen.mpg.de/algsysbio10/clustering.pdf Consultada el 18 de Junio

[10] Catherine A. Sugar and Gareth M. James Marshall School of Business, University of Southern California http://www-bcf.usc.edu/~gareth/research/ratedist.pdf Consultada el 18 de Junio

[11] Anil K. Jain Department of Computer Science Michigan State University https://pdfs.semanticscholar.org/1823/98ca4d85c25c64ba238dad10cafc92203660.pdf Consultada el 18 de Junio

[12] Página web de ffmpeg https://www.ffmpeg.org/documentation.html Consultada el 19 de Junio de 2018.

[13] https://docs.python.org/2/library/os.html Consultada el 19 de Junio de 2018.

[14] Página oficial de OPenCV https://opencv.org/ Consultada el 19 de Junio de 2018.

[15] Imagen extraida de: Use and its relationship to health in college students from the city of Manizales (Caldas-Colombia), 2015-2016 http://www.redalyc.org/jatsRepo/2738/273849945010/html/index.html

[16] VSUMM dataset. https://sites.google.com/site/vsummsite/home (https://www.dropbox.com/s/g0e64b4qfnuual1/database.zip) Consultada el 19 de Junio de 2018.

[17] Ph.D Ricardo Moya para Jarroba, página de difusión de nuevas tecnologías y colaboradora en el libro Machine Learning (en Python), con ejemplos https://jarroba.com/wp-content/uploads/2016/06/Cluster3C.png Consultada el 19 de Junio de 2018

[18] Visualización de clusters k-medias: https://www.naftaliharris.com/blog/visualizing-k-means-clustering/ Consultada el 20 de Junio de 2018.

[19] https://docs.opencv.org/2.4/doc/tutorials/imgproc/histograms/histogram_equalization/histogram_equalization.html Consultada el 18 de Junio de 2018.

[20] https://docs.opencv.org/2.4/doc/tutorials/imgproc/threshold/threshold.html Consultada el 18 de Junio de 2018.

[21] https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_imgproc/py_transforms/py_fourier_transform/py_fourier_transform.html Consultada el 18 de Junio de 2018.

[22] Página de difusión de nuevas tecnologías Jarroba https://jarroba.com/seleccion-del-numero-optimo-clusters/ Consultada el 18 de Junio de 2018.