Estudio y desarrollo de métodos para comparar vídeos

Eduardo Troconis Abreu

Resumen— Este trabajo tiene como objetivo llevar a cabo una investigación exhaustiva sobre las técnicas utilizadas para comparar el contenido audiovisual con el fin de mejorar un método existente o desarrollar uno nuevo. Se presenta una función de hash perceptual de vídeo basada en la extracción de una cadena binaria fija del contenido perceptual del material audiovisual. La función debe ser robusta a las operaciones de preservación del contenido y sensible a las modificaciones que varíen los contenidos perceptuales. Para ello, se emplea la Transformada Wavelet Temporal (TWT) obtenida a partir de la Transformada Wavelet Discreta a lo largo de la dirección temporal. La TWT permite generar Imágenes Representativas Temporalmente Informativas (TIRI) que son utilizadas como base para el cálculo del hash perceptual. El algoritmo extrae subtensores tridimensionales aleatorios de los coeficientes de baja frecuencia para generar el hash perceptual. Los resultados de la evaluación muestran que el algoritmo propuesto es eficaz tanto en situaciones en las que se busca preservar el contenido como en aquellas en las que se realizan cambios en él, además de tener una buena eficiencia computacional.

Palabras clave—Hashing perceptual de vídeo - Transformada wavelet temporal - Indexación y recuperación de vídeos casi idénticos – Estudio de métodos para comparar vídeos.

Abstract— This paper aims to carry out a comprehensive investigation into the techniques used to compare audiovisual content to improve an existing method or develop a new one. A video perceptual hash function based on the extraction of a fixed binary string from the perceptual content of audiovisual material is presented. The function must be robust to content preservation operations and sensitive to modifications that vary the perceptual contents. For this purpose, the Temporal Wavelet Transform (TWT) obtained from the Discrete Wavelet Transform along the temporal direction is used. The TWT allows the generation of Temporally Informative Representative Images (TIRI) that are used as the basis for the calculation of the perceptual hash. The algorithm extracts random three-dimensional subtensors from the low-frequency coefficients to generate the perceptual hash. The evaluation results show that the proposed algorithm is effective in both content-preserving and content-changing situations and has good computational efficiency.

Index Terms— Perceptual video hashing - Temporal wavelet transform - Indexing and retrieval of nearly identical videos - Study of methods to compare videos.

1 Introducción

El desarrollo de algoritmos enfocados en la comparación eficiente de contenido audio visual es cada vez más importante debido al auge de las redes sociales (por ejemplo, Instagram, Facebook, YouTube, TikTok, entre otras). Los usuarios de estas redes sociales comparten sus imágenes y vídeos o descargan una copia que puede ser editada para compartirla nuevamente. El hash de vídeo es una tecnología útil para este propósito entre otros como marcas de agua digitales, detección de copias y autenticación de contenido. En consecuencia, hay una gran cantidad de imágenes/vídeos en el ciberespacio [1], lo que representa un desafío detectar copias similares de imágenes/vídeos masivos [2, 3]. Para manejar estos problemas, muchos investigadores propusieron utilizar algoritmos hash [4-9] para procesar eficientemente datos multimedia.

La idea básica de las marcas de agua digitales es incorporar información de identificación en las secuencias de vídeo, reduciendo así los costos computacionales y haciendo que las marcas de agua digitales se utilicen ampliamente. Sin embargo, la detección de copia basada en marcas de agua no funciona siempre con transformaciones como rotar, desenfocar, recortar y escalar. Incluso los pequeños cambios provocados por las marcas de agua degradan la calidad del contenido, lo que lo hace inadecuado para algunas aplicaciones, que pueden precisar mucha resolución exacta. Si la versión original del vídeo se cargó en esas webs para compartir vídeos antes de que se insertara la marca de agua, la detección de duplicados basada en marcas de agua no tendrá ningún efecto.

Para subsanar esto se proponen funciones transformaciones sobre el audiovisual. Una función que genera un vector de características a partir del contenido perceptual del vídeo de entrada se denomina función hash de vídeo perceptual y el vector de características de salida que caracteriza el contenido perceptivo del vídeo de entrada se llama hash de vídeo perceptual. Este hash debe ser robusto a las manipulaciones que preserven los contenidos perceptuales del vídeo y frágil a las modificaciones que varíen los

- *E-mail de contacto: EduardoArturo.Troconis@autonoma.cat*
- *Mención realizada: Computación*
- *Trabajo tutorizado per: Jordi Serra Ruiz (Ciencias de la Computación)*
- *Curso 2022/23*

contenidos perceptuales del vídeo [10]. De esta manera podemos identificar un vídeo, aunque ya sea público.

Para evitar la detección, se pueden aplicar ciertas distorsiones al vídeo original, como distorsión geométrica (por ejemplo, rotar o escalar) y distorsión de retención de contenido (por ejemplo, compresión con pérdida o mejora del contraste), y crear así una copia del vídeo. Estas distorsiones comunes se pueden clasificar en tres categorías: transformaciones geométricas (como rotación, volteo y escalado), transformaciones espaciales (como ajuste de contraste, adición de ruido y desenfoque) y transformaciones temporales (como cambiar la velocidad de fotogramas).

Las duplicaciones de vídeo son clips de vídeo que se manipulan o transforman para que sean similares o menos similares, pero no idénticos, al vídeo original [11, 12]. Debido a la amplia variedad de transformaciones y grandes volúmenes de datos, sigue siendo una tarea difícil utilizar un método rápido y sólido para detectar con precisión copias ilegales del vídeo original.

Según los métodos de extracción de características, los hashes de vídeo robustos se pueden dividir en tres categorías principales: espaciales, temporales y espaciotemporales.

El algoritmo de hash basado en características espaciales trata el vídeo como una colección de fotogramas y extrae las características de la imagen de cada fotograma. Dado que el hash de imágenes es geométricamente invariante, los métodos de hash basados en características espaciales muestran su solidez tanto en el procesamiento de vídeo convencional como en las transformaciones geométricas. Sin embargo, todos los métodos de hash basados en características espaciales carecen de robustez frente a las transformaciones temporales, ya que se ignora la información temporal del vídeo. Además, las búsquedas en bases de datos son costosas debido a él gran tamaño de los valores hash generados por las características espaciales.

La información de vídeo temporal representa contenido visual que cambia con el tiempo y es la clave para distinguir entre vídeo dinámico e imágenes fijas. Las características temporales están más cerca de los conceptos semánticos que las características espaciales. Los clips cortos no tienen suficiente información de tiempo identificable, por lo que generalmente funcionan bien con secuencias de vídeo largas. No obstante, los métodos hash basados en características temporales no son robustos a las transformaciones geométricas. Por último, encontramos la información espacial y temporal del vídeo en el hash basado en características espaciotemporales. Algunos investigadores utilizan los coeficientes de baja frecuencia de la Transformada de coseno discreta tridimensional (3D-DCT) o la transformada wavelet discreta 3D (3D-DWT) para generar el hash de un vídeo. Si bien estos métodos son robustos para transformaciones espaciales y temporales, no son adecuados para transformaciones geométricas.

Con el avance del aprendizaje profundo, las redes neuronales profundas (DNN) han mejorado significativamente en el campo de la visión por computador. Los investigadores han utilizado DNN para extraer características del vídeo, dado que las DNN tienen la capacidad de integrar información en áreas y fotogramas adyacentes, lo que permite un rendimiento notable. Sin embargo, también existen desventajas en el uso de DNN, como la necesidad de entrenamiento que conlleva un alto costo computacional y la falta de robustez ante transformaciones como la rotación y adición de ruido.

El resto del trabajo está organizado de la siguiente forma. En la sección 2 se realiza un estudio exhaustivo sobre los métodos utilizados para comparar el contenido audiovisual. La sección 3 presenta los métodos que resultaron interesantes y el algoritmo de hash de vídeo propuesto para la detección de copias de vídeo. Los resultados experimentales se describen en la sección 4, en la que comparamos el hash de vídeo propuesto con el método de Mahanty [37]. En la sección 5 se presenta una conclusión.

1.1 Objetivos

Este trabajo tiene como objetivo llevar a cabo una investigación exhaustiva sobre las técnicas utilizadas para comparar el contenido audiovisual, con el fin de implementar el método más interesante para proponer mejoras o desarrollar un nuevo método.

2 Estado del arte

El método de detección de copias de vídeo basado en hash determina principalmente si el contenido sospechoso es similar al documento multimedia registrado en la base de datos de hash mediante la extracción de un código hash compacto y el uso de una métrica de distancia. En esta sección, se describen varios métodos existentes de detección de copias de vídeo basados en hash, que se clasifican en métodos basados en características espaciales, métodos basados en características temporales, métodos basados en características espaciotemporales y métodos basados en redes neuronales profundas dependiendo de la diferencia en la etapa de extracción de características.

2.1 Métodos basados en características espaciales

Los métodos basados en características espaciales extraen el código hash o vector de características de cada fotograma clave o de cada fotograma de un vídeo. Algunos investigadores han utilizado descriptores locales para formar un vector de código hash compacto.

Zhang et al. [13] han introducido un método que es invariable a la rotación basado en SURF (Características Robustas Aceleradas) para extraer características locales de cada fotograma y utiliza el hash local sensible (LSH) para mejorar el rendimiento y generar un hash compacto. El alto costo computacional es el inconveniente de este método y también es menos sensible a los cambios globales, como la variación de color.

Lee et al. [14] han propuesto un método de hash de vídeo en el que se extraen las características globales haciendo uso del Centroide de Orientaciones de Gradiente (CGO). Cada fotograma es redimensionado y dividido en bloques para calcular el CGO de cada bloque. Este método es robusto contra ciertas distorsiones, como compresión con pérdida, cambio de velocidad de fotogramas, pero no lo es ante transformaciones geométricas generales como la rotación y el recorte.

Hua et al. [15] utilizaron la medida ordinal (OM) para generar el hash de vídeo. Cada fotograma se divide en un número de bloques, para cada bloque, se calcula el valor de gris promedio y se clasifican en orden creciente. Generalmente, la medida ordinal es robusta frente a transformaciones como ruido, filtrado, compresión y a la degradación del color. Sin embargo, no lo es ante las transformaciones locales, como la inserción de logotipos, recortes, etc.

Su et al. [16], extrajeron el vector de características de las regiones de atención visual que estaban representadas por un mapa de prominencia. El mapa de prominencia único se formó mediante la combinación de mapas de características visuales normalizados, como mapas de color, mapas de intensidad y mapas de orientación, que se calcularon a partir del fotograma de entrada. Por último, el mapa de prominencia se divide en M x N bloques y se calcula el valor promedio. Todos los valores de los bloques se cuantifican de forma adaptativa como hash de vídeo. Se utilizó el enfoque ascendente, que evita el efecto del nivel superior del sistema visual humano (HVS). Este método es robusto frente a las distorsiones que preservan el contenido, pero no frente a las distorsiones geométricas.

Zheng et al. [17], basándose en un concepto similar a [16] han propuesto un algoritmo de covarianza prominente (SCOV) para la detección de imágenes y vídeos casi idénticos. Para generar un descriptor compacto y robusto utilizan la matriz de covarianza de características de imagen visualmente prominentes. Calculan la distancia entre las características prominentes de dos fotogramas y con la detección de líneas a través de la transformada de Hough obtienen los segmentos similares en un vídeo. El alto costo computacional y la menor capacidad discriminativa son las principales desventajas de este método.

Sarkar et al. [18], han utilizado el descriptor de diseño de color (CLD), que se obtiene redimensionando el fotograma a 8×8, en promedio, a lo largo de cada canal (YCbCr). La transformada de coseno discreta (DCT) se calcula para cada fotograma. Los coeficientes DC y los cinco primeros (exploración en zigzag) AC para cada canal constituyen la característica CLD de 18 dimensiones. La función CLD captura el contenido de frecuencia en una representación muy gruesa del cuadro y se codifica aún más mediante la cuantificación vectorial (VQ). La desventaja de este enfoque es que cambios significativos en el brillo y el recorte pueden afectar al CLD y causar errores.

En [19] Sun et al. presentaron un nuevo método basado en el modelo de árbol de Markov oculto por transformación de contorno (CHMT) y la descomposición de valores singulares (SVD). En este método, cada fotograma es redimensionado y dividido en M × N bloques para luego aplicar la transformación de contorno y capturar contornos suaves que son las características dominantes en las imágenes naturales. El modelo CHMT captura las dependencias entre escalas, direcciones y ubicaciones de los coeficientes de contorno usando pocos parámetros estadísticos. Por último, la descomposición de valores singulares es utilizada para reducir la dimensión de las matrices de desviación estándar y se selecciona como vector de características el mayor valor singular de cada matriz. Este método robusto frente a operaciones comunes de conservación de contenido como la compresión con pérdida, filtrado, pero no frente a los ataques geométricos.

Gu et al. [20] han adoptado el descriptor SIFT [21] para la extracción de características locales para estimar la transformación de la copia, y luego, se utilizó la medida ordinal [15] como una característica global para acelerar la detección de copias posteriormente. Se usó el algoritmo de consenso de muestra aleatoria (RANSAC) para estimar la transformación afín que asigna los puntos en el fotograma de consulta a los de su fotograma de referencia coincidente. La transformación afín puede modelar los cambios geométricos introducidos por las transformaciones, como imagen sobre imagen, desplazamiento, zoom, etc. Se ha aprovechado para descartar los puntos de características locales que no coinciden. Este método presenta un alto costo computacional.

Los métodos de hash basados en características espaciales anteriores no son robustos frente a transformaciones temporales como el cambio de velocidad de fotogramas debido a que no tienen en cuenta la información temporal del vídeo. La selección de fotogramas clave para representar de manera eficiente el vídeo es un problema importante en estos métodos. Además, la alta dimensión del hash generado a partir de características espaciales hace que la búsqueda en la base de datos sea computacionalmente costosa.

2.2 Métodos basados en características temporales

Los métodos basados en características temporales utilizan valores hash entre varios fotogramas consecutivos en la dirección temporal para describir un vídeo

Chen y Stentiford [22] proponen un método de emparejamiento de secuencias de vídeo basado en mediciones ordinales como en [15] pero extendido al dominio temporal. Cada fotograma se divide en bloques y los bloques correspondientes a lo largo del tiempo se ordenan en una secuencia de clasificación ordinal, lo que brinda una descripción global y local de la variación temporal.

La medida ordinal proporciona robustez y discriminabilidad frente a ciertas transformaciones, como el cambio en la tasa de fotogramas, la compresión, etc., pero no puede tolerar transformaciones que cambien un subconjunto de fotogramas en el vídeoclip, como el recorte de regiones.

Radhakrishnan y Bauer [23] presentaron un nuevo método de hash de vídeo basado en la proyección subespacial. Se divide el vídeo en intervalos para formar grupos de fotogramas y usando una ventana deslizante, se extraen las características de cada grupo. Para obtener las características, primero se calculan los vectores base de una representación aproximada de cada grupo utilizando la Descomposición de Valor Singular (SVD). Luego, se proyecta la representación aproximada en un subconjunto de los vectores base y se obtiene una representación subespacial de los fotogramas de vídeo de entrada. Finalmente, el hash se generó proyectando un promedio temporal de las representaciones en vectores de base pseudoaleatoria. Este método es robusto en ciertas transformaciones como el cambio en la tasa de fotogramas, compresión, escalado espacial pero no es resistente a ciertas transformaciones, como cambios

de iluminación o recorte.

Tasdemir y Enis en [24], utilizaron la combinación de la media de las magnitudes de los vectores de movimiento (MMMV) y la media de los ángulos de fase de los vectores de movimiento (MPMV) para extraer los vectores de movimiento a lo largo de la dirección temporal. Este método no produce resultados precisos cuando los vectores de movimiento se extraen de fotogramas consecutivos con una alta tasa de captura.

Zhang et al. [25] generaron el hash de vídeo a través del cálculo de histogramas de orientaciones de flujo óptico de puntos característicos obtenidos de los fotogramas muestreados de manera uniforme a lo largo del tiempo. Estas series temporales son luego alineadas y combinadas. Además, se utiliza la técnica de coincidencia de inclinación principal, como un método de reducción de datos y alineación de picos, para mejorar el rendimiento. Es un método compacto y resistente a diversas transformaciones como voltear, recortar, superponer imágenes o adición de ruido.

Aunque las funciones temporales suelen ser efectivas para secuencias de vídeo largas, presentan algunas desventajas al utilizar solo información temporal para la detección de copias de vídeo: (1) no se pueden aplicar a segmentos de vídeo de duración corta, solo a vídeos de duración larga; (2) no son robustas frente a transformaciones geométricas en el vídeo.

2.3 Métodos basados en características espaciotemporales

Los métodos basados en características espaciotemporales utilizan la información tanto espacial como temporal de un vídeo.

Coskun et al. [7] aplicaron la transformada de coseno discreta tridimensional (3D-DCT) a la componente de luminancia del vídeo. Los coeficientes de baja frecuencia se ordenaron y cuantificaron usando la mediana de los coeficientes ordenados por rango, generando $4 \times 4 \times 4$ bits binarios para cada cubo 3D y obtener el hash del vídeo. Este método es robusto frente transformaciones como compresión, cambios en el contraste o en la tasa de fotogramas, pero no lo es frente a la inserción de fotogramas o imagen sobre imagen.

Esmaeili et al. [26] generó una imagen representativa informativa temporal (TIRI) a partir de los fotogramas del vídeo mediante el cálculo de una suma ponderada de fotogramas, aplicó DCT 2D a bloques superpuestos de cada TIRI y seleccionó los primeros coeficientes verticales y horizontales para construir hash. El esquema TIRI-DCT es resistente al ruido y la caída de tasa de fotogramas.

Sun et al. [27] combinó la característica de apariencia visual extraída de cada bloque de los fotogramas temporales representativos (TRF), y la característica de atención visual extraída de cada bloque de los mapas de prominencia representativos (RSM), a través de una red de creencia profunda (DBN) para obtener el valor de hash compacto que representa todo el vídeo. Sin embargo, no es robusto frente a la inserción de fotogramas o el recorte.

Nie et al. [28] introdujo una técnica de proyección basada en un modelo de tensor de alto orden al que extrajo la asistencia y el consenso entre diferentes características para obtener el hash del vídeo. El tensor de alto orden se descompone mediante el modelo de Tucker para generar a partir del tensor de orden bajo obtenido, una característica completa con la que se creará el hash de vídeo. Las proyecciones basadas en tensores son robustas capturando las características espaciotemporales del vídeo, sin embargo, la inserción aleatoria fotogramas y un gran cambio en el brillo puede afectar a este método.

Douze et al. [29] calculan el hash del vídeo a partir de un subconjunto de fotogramas, ya sea muestreados periódicamente de la secuencia de vídeo o elegidos de acuerdo con una regla de contenido visual (fotogramas clave). En este modelo, el cambio temporal se determinó primero en función de la estrategia de votación de Hough unidimensional y la componente espacial se determinó mediante la estimación de la transformación afín bidimensional entre las secuencias de vídeo coincidentes.

La información visual local se extrae a través de detectores Hessian-Affine seguidos de descriptores SIFT y CS-LBP [21,30]. Posteriormente, los descriptores se agrupan mediante un enfoque de bolsa de palabras combinado con un procedimiento de incrustación de Hamming. Este método es robusto frente a la reducción en la tasa de fotogramas y al recorte.

Kim et al. [31] han propuesto un método que combina la información espacial y temporal de una secuencia de vídeo. El hash de vídeo espacial se extrajo de los signos de los coeficientes DCT en áreas locales en un fotograma clave y el hash de vídeo temporal se extrajo usando las varianzas temporales en áreas locales en fotogramas clave consecutivos. Por último, la técnica de medición de la fuerza temporal permite cuantificar la cantidad de variaciones temporales y se puede utilizar de forma adaptativa para evaluar la importancia de las huellas dactilares temporales. Este método es robusto frente al desenfoque, el cambio de brillo y el cambio de velocidad de fotogramas, y también contra el recorte, inserción de subtítulos y volteo.

Rameshnath y Bora [32] han utilizado recientemente una combinación de la matriz aleatoria (ARM) de Achlioptas y la transformada de onda temporal (TWT) para crear un método de hash de vídeo que es resistente a ataques como cambios de brillo fuertes, desenfoque gaussiano y la inserción de marcas de agua.

Los algoritmos de hash basados en características espaciotemporales son robustos frente a transformaciones temporales de vídeo, como la pérdida de fotogramas y el cambio de velocidad de fotogramas, al mismo tiempo que mantienen su robustez frente a transformaciones espaciales. Sin embargo, muchos métodos carecen de robustez frente a transformaciones geométricas.

2.4 Métodos basados en redes neuronales profundas

Los métodos basados en redes neuronales profundas han demostrado ser efectivos en la extracción de características y han sido utilizados con mayor frecuencia en la investigación.

Un enfoque, propuesto por Hu y Lu [33], utiliza tanto una red neuronal convolucional (CNN) como una red neuronal recurrente (RNN) para obtener una mayor precisión

en la detección de copias. La red neuronal convolucional residual (ResNet) extrae las características de los fotogramas y, a continuación, se entrenó una arquitectura de memoria siamesa a corto plazo (SiameseLSTM) para fusionar las características espaciales y temporales y para establecer la correspondencia de las secuencias de vídeo. Finalmente, se empleó una red neuronal basada en grafos para identificar los segmentos copiados en un vídeo.

En el trabajo de Li et al. [34], se presenta un enfoque de red neuronal convolucional tridimensional paralela (3D-CNN) para la clasificación de vídeo. Utiliza varias 3D-CNN para extraer características directamente del flujo de vídeo de entrada y obtener la información de movimiento local. Sin embargo, este enfoque puede resultar en altos costos computacionales y no se ha evaluado adecuadamente su resistencia a ataques como la transformación geométrica y las transformaciones que preservan el contenido.

En este trabajo, Anuranji y Srimathi [35] proponen un modelo de red conjunta supervisado de múltiples núcleos convolucionales heterogéneos apilados (Stacked HetConvMK)-bidireccionales de memoria a largo y corto plazo (BiDLSTM) para codificar las características estructurales y discriminativas del vídeo. Inicialmente, los fotogramas de vídeo se pasan a las redes de convolución apiladas con un tamaño de núcleo convolucional heterogéneo y aprendizaje residual para extraer las características espaciales representativas. A continuación, la red bidireccional mejora la eficacia de la codificación. Por último, una capa totalmente conectada genera una huella binaria que integra la salida de las unidades anteriores.

En Nie et al. [36] proponen un método de hashing de vídeo denominado hashing profundo de mejora de la clasificación (CEDH). CEDH es un modelo de aprendizaje profundo que se compone de 3 capas principales. En primer lugar, una capa VGGNet-19 para extraer características semánticas a nivel de fotograma. A continuación, se adopta una red LSTM para capturar características temporales. Por último, se implementa un módulo de clasificación para mejorar la información de la etiqueta. Para entrenar el modelo, el término de pérdida se ajusta a las peculiaridades de la capa: pérdida de triplete, pérdida de clasificación y términos de restricción de código.

La mayoría de los métodos que se basan en redes neuronales profundas tienen la capacidad de generar características sólidas y obtener alta precisión en los resultados, pero suelen carecer adaptabilidad y robustez. Esto implica que deben ser específicamente entrenados para manejar ciertas transformaciones para lograr resultados satisfactorios. Además, el proceso de entrenamiento conlleva un alto costo computacionales y una gran cantidad de datos de entrenamiento son necesarios.

3 METODOLOGÍA

El objetivo de este trabajo es desarrollar un método capaz de extraer las características temporales y espaciales de las secuencias de vídeo y generar un hash robusto para la detección de copias de vídeo. Con el propósito de desarrollar este método, se ha llevado a cabo una investigación exhaustiva sobre las distintas estrategias utilizadas para efectuar la comparación de vídeos, así como el software disponible.

A fin de comprender mejor el funcionamiento de la metodología para comparar vídeos se ha seleccionado el software creado por Mahanty [37], en el cual se construye un collage con los fotogramas del vídeo y se aplica 2D-DWT para obtener las características espaciales. Además, se detecta el color dominante de estas imágenes y se compara con un patrón predefinido. Por último, con las características extraídas genera el hash del vídeo. El análisis del código y funcionamiento del método propuesto por Mahanty nos proporcionó un marco de referencia para desarrollar nuestro método.

Se optó por implementar aquellos trabajos cuyo enfoque y metodología se ajusten mejor a nuestro objetivo de investigación. Primero se aplicó el método propuesto por Chen, Wo y Han [38], donde se aplica la técnica de transformación exponencial compleja polar (PCET) en la subbanda de paso bajo de la transformada de onda discreta 3D (3D-DWT) en la secuencia de cuadros para obtener el hash espaciotemporal geométrico invariable, el cual se utiliza para la autenticación de vídeo. Los momentos PCET locales se calculan en bloques anulares y angulares, los cuales se emplean para la corrección geométrica y la localización de manipulaciones gruesas. Además, se utiliza el mapa de prominencia para obtener la información de posición de los objetos destacados y así lograr la localización de manipulaciones finas. Sin embargo, durante la implementación de esta técnica, hubo problemas calculando la transformación PCET en la subbanda de paso bajo y en la obtención de los momentos PCET locales calculados en bloques anulares y angulares.

Del método anterior [38], la transformada de onda discreta demostró ser robusta frente a la mayoría de los ataques de procesamiento de imágenes individuales y múltiples. Por esta razón se decidió implementar el método propuesto por Rameshnath y Bora [32], el artículo presentó una combinación de la matriz aleatoria (ARM) de Achlioptas y la transformada wavelet temporal (TWT) para crear un método de hash de vídeo. Se implementó el método TWT-ARM con algunas modificaciones, ya que algunos pasos presentaban una explicación confusa. Se encontró que al redimensionar los fotogramas, insertar fotogramas hasta que sean igual a la potencia de dos más cercana y aplicar TWT hasta cierto nivel de descomposición dependiendo del número de fotogramas, la dimensión temporal debía ser igual a la dimensión espacial para obtener el vídeo normalizado a $X \times X \times X$. Debido a que al aumentar la dimensión espacial a más de 256 aumentaba considerablemente el tiempo de ejecución, se decidió no utilizar una dimensión de fotogramas superior a 256 ni un nivel de descomposición mayor que dos. En caso de que se interprete que se retienen cierto número de fotogramas para generar el hash del vídeo y la dimensión temporal posea más de 256 fotogramas, se perderían las características espaciotemporales del final del vídeo. Por lo que se optó por no retener solo los primeros fotogramas y crear los subtensores 3D tomando en cuenta toda la dimensión temporal. Otro inconveniente surgió con el vector de hash intermedio, es construido a partir de la media de las columnas de

la matriz. Al hacer nuestras pruebas no daba buen resultado, en cambio al realizar la media de las filas los resultados mejoraron. Se tuvo en consideración distinto orden al concatenar los elementos de los subtensores para formar el vector de características.

Nuestro objetivo inicial es identificar características que sean robustas a las transformaciones espaciotemporales como a las transformaciones geométricas. Con base en el estudio realizado y en el desarrollo del método propuesto por Rameshnath y Bora [32], se ha determinado que la mejor forma de lograr esto es mediante el uso de la Transformada Wavelet Discreta en nuestro método. El algoritmo consta de los siguientes bloques: preprocesamiento y normalización del vídeo, selección de la orientación temporal y espacial del vídeo, TWT, selección pseudoaleatoria de subtensores y cálculo del hash. El algoritmo genera la función hash capturando la esencia espaciotemporal del vídeo de entrada, y, por tanto, se clasifica dentro de la categoría de algoritmos de hashing de vídeo espaciotemporal. A continuación, se describirán de manera concisa los componentes mencionados.

3.1 Preprocesamiento y normalización

Dado que el tamaño del hash generado es constante, se procede a normalizar dimensión espacial de los vídeos a $X \times X$ mediante interpolación bilineal. La dimensión temporal también es normalizada a $2 \times X$ mediante interpolación lineal. Además, se convierte la representación del vídeo en el espacio de color RGB a un nivel de grises, con el fin de minimizar el impacto de las diferencias en las dimensiones espaciales y la crominancia en el valor del hash. El resultado de este proceso de normalización se representa con la notación V_{norm}.

3.2 Selección de la orientación temporal y espacial del vídeo

Uno de los ataques más comunes es voltear en el eje vertical el vídeo. Para evitar este ataque, se compara la suma de un conjunto de pixeles del lado izquierdo de los fotogramas de V_{norm} con la suma de un conjunto de pixeles del lado izquierdo de los fotogramas invertidos en el eje vertical para obtener la orientación espacial. Además, se aplica este mismo método, pero en un lado temporal del vídeo para obtener la orientación temporal. Por último, se aplica la orientación obtenida de los fotogramas en V_{norm}.

3.3 Transformada Wavelet Temporal

La transformada wavelet discreta aplicada en tres dimensiones es una transformada que descompone la señal del vídeo en varios conjuntos, donde cada conjunto es una serie temporal de coeficientes que describen la evolución temporal de la señal en la banda de frecuencias correspondiente. Debido a este motivo, se aplica la Transformada Wavelet Temporal (TWT) sobre la señal normalizada V_{norm}, utilizando una wavelet de Haar. Se considerarán los coeficientes de baja frecuencia, ya que estos proporcionan información relevante localizada en el tiempo y el espacio. El nivel de descomposición se determina en función de la dimensión espaciotemporal del vídeo y la dimensión requerida en los coeficientes de baja frecuencia. Los fotogramas de baja frecuencia del vídeo normalizado forman $V_{norm-trans}$. Por ejemplo, supongamos que nuestro objetivo es obtener los coeficientes de baja frecuencia con una dimensión de $64 \times 64 \times 128$. Si la dimensión espacial es de 128×128 y la dimensión temporal es de 2×128, entonces se requiere aplicar un nivel de descomposición. Esto se debe a que, en cada nivel de descomposición, la dimensión espacial y temporal se divide entre 2^l siendo l el nivel de descomposicion.

3.4 Selección pseudoaleatoria de subtensores

Con base en una clave secreta K, los N subtensores 3D se seleccionan de forma pseudoaleatoria cubriendo aproximadamente todo $V_{norm-trans}$ para formar $V_i = 1, 2, \ldots, N$, cada uno de tamaño $U \times U \times U$ donde $U = X/4$. La clave secreta garantiza la seguridad en la generación del hash, la selección de los subtensores 3D para una clave diferente, tendrá como resultado un hash diferente. Cada subtensor 3D se reordena para formar una columna de la matriz de características, $\mathbf{F} = [\mathrm{f}_1, \mathrm{f}_2, \ldots, \mathrm{f}_N]$. Esta matriz captura tanto la información local del vídeo normalizado en cada columna como la información global en su conjunto. La robustez ante ataques geométricos es una ventaja de $\mathbf{F}$, ya que aun cuando se pierdan partes de los fotogramas originales en un ataque, esto sólo afectará a pocos componentes de la matriz de características y no tendrá un impacto significativo en la información global.

3.5 Cálculo del hash

El promedio aritmético se realiza sobre los datos de la matriz de características para obtener un vector hash intermedio,

$$h' = \frac{\sum_{i=1}^{d} h(:,i)}{d} \tag{1}$$

donde $h(:,i)$ es la i-ésima columna de la matriz de características $\boldsymbol{F}$. El promedio mejora la robustez y reduce la longitud del hash de vídeo. Por último, el hash de vídeo h se obtiene binarizando los elementos del vector hash intermedio utilizando la mediana ψ, de los elementos del vector hash intermedio como umbral. Matemáticamente,

$$h_j = \begin{cases} 0; h'_j < \psi \\ 1; h'_j \geq \psi \end{cases}, j = 1, 2, \ldots, N. \tag{2}$$

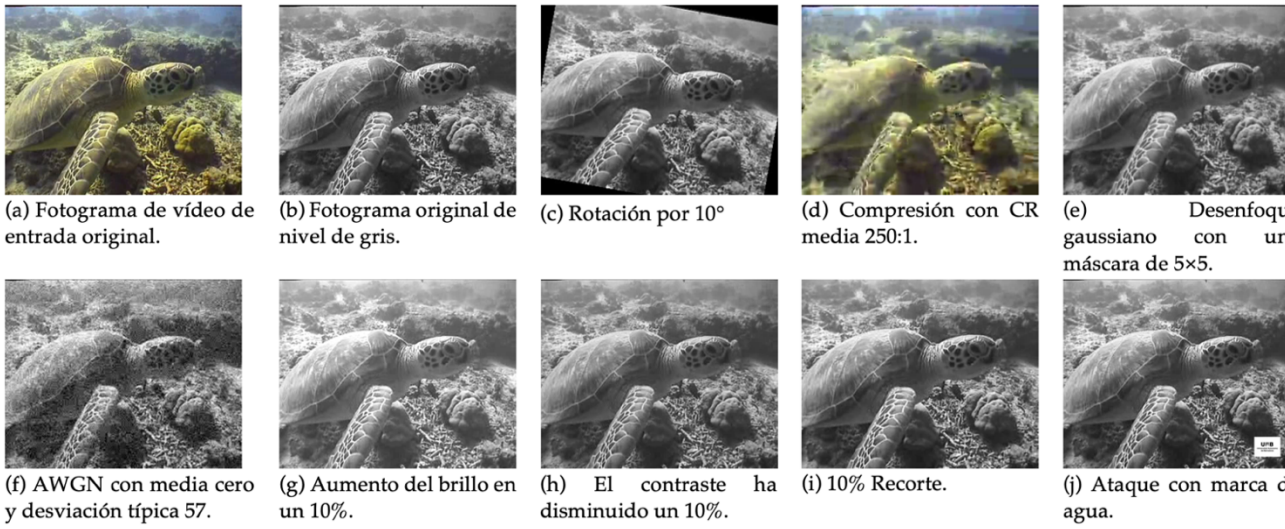
donde h'_j y h_j son los j-ésimos elementos de h' y h, respectivamente.

Los pasos algorítmicos del método hash propuesto se resumen en el Algoritmo 1.

Algoritmo 1 El algoritmo propuesto

Entrada: RGB vídeo V_{in}; Parámetros: X, N and K
Salida: Un vector hash **h**.

1. Se transforma V_{in} en una secuencia de fotogramas a nivel de gris V_{gris}
2. Cada fotograma se redimensiona a $X \times X$ mediante interpolación bilineal.
3. El número de fotogramas se ajusta a $2 \times X$ mediante interpolación lineal obteniendo V_{norm}
4. Se compara la suma de un conjunto de pixeles del lado izquierdo de los fotogramas de V_{norm} con la suma de un conjunto de pixeles del lado izquierdo de los fotogramas invertidos en el eje vertical para



(a) Fotograma de vídeo de entrada original. (b) Fotograma original de nivel de gris. (c) Rotación por 10° (d) Compresión con CR media 250:1. (e) Desenfoque gaussiano con una máscara de 5×5.

(f) AWGN con media cero y desviación típica 57. (g) Aumento del brillo en un 10%. (h) El contraste ha disminuido un 10%. (i) 10% Recorte. (j) Ataque con marca de agua.

Fig. 1 Ilustraciones de fotogramas de nivel de gris bajo diversos ataques considerados para los experimentos

aplicar la orientación de los fotogramas en V_{norm}

5 Se compara la suma de un conjunto de pixeles en un lado temporal de los fotogramas de V_{norm} con la suma un conjunto de pixeles en un lado temporal de los fotogramas invertidos en el eje temporal para aplicar la orientación temporal en V_{norm}

6 Se aplica TWT en V_{norm} hasta un nivel l-ésimo de descomposición, los fotogramas de baja frecuencia obtenidos forman $V_{norm-trans}$

7 En función de la longitud hash N y de la clave K. Los N subtensores 3D se seleccionan de forma pseudoaleatoria de $V_{norm-trans}$ para formar $V_i = 1, 2, \dots, N$, cada uno de tamaño $U \times U \times U$ donde $U = X/4$.

8 Se concatenan las columnas de cada subtensor Vi para formar un vector $f_i \in \mathbb{R}^d$ tal que d=$U \times U \times U$

9 Se construye la matriz de características $\mathbf{F} \in \mathbb{R}^{N \times d}$ de los vectores f_i para obtener $\mathbf{F} = [f_1, f_2, \dots, f_N]$

10 Se genera el vector intermedio del hash $h' = \frac{\sum_{i=1}^{d} h(:,i)}{d}$, donde $h(:,i)$ es la i-ésima columna de la matriz $\boldsymbol{F}$.

11 Se ordena h' para obtener $h'' = [h'_{(1)}, h'_{(2)}, \dots, h'_{(N)}]$

12 Se obtiene la mediana ψ de $h'' = \begin{cases} \left(\frac{h''_{(\frac{N}{2})} + h''_{(\frac{N}{2}+1)}}{2} \right); \text{si } N \text{ es par} \\ h''_{(\frac{N+1}{2})}; \; Si\ N\ es\ impar \end{cases}$

13 Se binarizan los elementos de h' para obtener $h = [h_1, h_2, \dots h_N]$ donde $h_j = \begin{cases} 0; h'_j < \psi \\ 1; h'_j \geq \psi \end{cases}, j = 1, 2, \dots, N.$

4 Resultados experimentales

En esta sección, analizamos el rendimiento del método de hashing robusto de propuesto a través de una serie de experimentos. Nuestro método se compara con el método de Mahanty [37], que hace uso de la transformada wavelet discreta (DWT). Los experimentos se realizan en Python 3.10, PC con Mac OS Monterey, CPU Intel Core i5-6267U a 2,9 GHz y 16GB de RAM. En el paso de preprocesamiento, cada vídeo se procesa a 128 × 128 × 256.

4.1 Ataques

Se realizarán pruebas contra varios tipos de ataques que preservan y modifican el contenido, ataques geométricos, ataques temporales y ataques espaciales. La Figura 1 ilustra, a manera de ejemplo, un fotograma de vídeo sometido a diferentes tipos de ataques evaluados en los experimentos. Estos ataques incluyen: rotación de fotogramas, compresión, caída de fotogramas, desenfoque gaussiano, adición de ruido, modificación del brillo, modificación del contraste, recorte de fotogramas, reproducción inversa, reproducción de vídeo invertido en el eje vertical e inserción de marcas de agua. Los ataques fueron se realizados con FFMPEG, ya que permite realizar tareas de edición de vídeo de forma eficiente y personalizada.

4.2 Conjunto de vídeos y evaluación

Para generar vídeos similares fueron seleccionados 78 vídeos de las bases de datos de vídeos abierta [39, 40]. Estos vídeos están en formato AVI y MP4, presentan varias resoluciones siendo la mínima 320×240 y la máxima 368×480, y sus números de fotograma oscilan entre 107 y 31490. En este experimento, el número de operaciones digitales utilizadas es de 13 y se seleccionan dos parámetros para la rotación 5° y 10°. Esto significa que existen 78 vídeos y 52 versiones similares de cuatro vídeos originales. Por tanto, el número total de vídeos es de 130. Para facilitar la comparación con el método de Mahanty [37], hemos adoptado los siguientes indicadores:

$$precision = \frac{TP}{TP+FP} \tag{3}$$

$$recall = \frac{TP}{TP+FN} \tag{4}$$

donde TP significa verdadero positivo, FP significa falso positivo y FN significa falso negativo. El índice de

precisión (*precision*) calcula la proporción de todos los resultados recuperados correctamente (TP) con respecto a todos los realmente recuperados (TP + FP), y el índice de exhaustividad (*recall*) calcula la proporción de todos los resultados recuperados correctamente (TP) con respecto a todos los resultados que deberían recuperarse ($TP + FN$). La combinación de precisión y exhaustividad puede distinguir con exactitud los dos errores siguientes: uno es juzgar la muestra positiva como negativa, y el otro es juzgar la muestra negativa como positiva. De este modo, la información sobre las muestras y la información sobre los resultados predichos pueden visualizarse completamente y el rendimiento del algoritmo puede medirse de forma más exhaustiva. En la detección de múltiples categorías de objetos, cada categoría puede dibujar una curva PR según *precision* y *recall*, mientras que la precisión media (AP) es el área bajo la curva, y la precisión media promedio (mAP) es la media de AP en múltiples categorías. Este valor permite una evaluación exhaustiva de los métodos, por lo que utilizamos la mAP para la evaluación final.

4.3 Comparación del rendimiento de la transformación geométrica y temporal

Evaluamos el rendimiento de los dos algoritmos ante transformaciones geométricas como rotación y escalado y transformaciones temporales como caída de fotogramas y reproducido al revés. Se puede observar en la Tabla 1 que el algoritmo propuesto y el algoritmo DWT muestran muy buen rendimiento con valores mAP iguales a 1. Excepto en la reproducción al revés, esta transformación afecta el flujo del tiempo del vídeo, y se realiza con el propósito de crear un efecto visual diferente.

Tabla 1 Comparación de los valores mAP entre la variación geométrica y la variación temporal

	Método Propuesto	DWT
Rotación 5°	1.0000	1.0000
Rotación 10°	1.0000	1.0000
Escalado	1.0000	1.0000
Caída de fotogramas	1.0000	1.0000
Tiempo invertido	1.0000	0.9743

Se puede concluir que el algoritmo propuesto es robusto tanto frente a cambios temporales como a transformaciones geométricas.

4.4 Comparación del rendimiento de las transformaciones espaciales

Evaluamos el rendimiento de los dos algoritmos ante transformaciones espaciales como tasa de compresión 250:1, desenfoque utilizando una máscara gaussiana de tamaño 5 × 5, adición del AWGN con media 0 y desviación típica 57, incremento del brillo 10%, reducción del contraste 10%, recorte del 10% del fotograma, marca de agua e inversión de los fotogramas en el eje vertical lo que resulta en una imagen reflejada. Se puede observar en la Tabla 2 que el algoritmo propuesto y el algoritmo DWT muestran muy buen rendimiento con valores mAP iguales a 1. Excepto cuando se invierte la orientación de las imágenes en el vídeo, una trasformación espacial muy común y simple de realizar.

Tabla 2 Comparación de los valores mAP según la variación espacial

	Método Propuesto	DWT
CR250:1	1.0000	1.0000
Desenfoque Gaussiano	1.0000	1.0000
AWGN	1.0000	1.0000
Brillo +10%	1.0000	1.0000
Contraste -10%	1.0000	1.0000
Recorte 10%	1.0000	1.0000
Marca de agua	1.0000	1.0000
Fotograma invertido	1.0000	0.9743

Se puede concluir que el algoritmo propuesto es robusto frente a transformaciones espaciales.

4.5 Velocidad de análisis

La Tabla 3 muestra la velocidad de análisis de los dos algoritmos hash, se obtiene dividiendo el número total de fotogramas por el tiempo total de análisis (en segundos). Esto nos da una idea de cuántos fotogramas puedes analizar por segundo. Se observar que nuestro algoritmo hash posee muy buena eficiencia computacional. El algoritmo DWT obtiene un tiempo mucho más elevado para analizar un vídeo debido al procesamiento por separado de cada fotograma.

Tabla 3 velocidad de análisis (unidad: fotogramas por segundo)

	Método Propuesto	DWT
Fotogramas	1776.7365	318.6622

5 Conclusión

En este trabajo, se lleva a cabo una investigación exhaustiva sobre los métodos utilizados para comparar el contenido audiovisual y se ha propuesto un algoritmo de hashing perceptual de vídeo basado en la Transformada de Wavelet Temporal (TWT). Las características perceptuales a lo largo de la dirección temporal se capturaron utilizando la TWT en los fotogramas de baja frecuencia. A partir de estos datos seleccionaron subtensores 3D aleatoriamente. El algoritmo hash de vídeo propuesto se comparó con el algoritmo hash de vídeo de Mahanty [37]. Mostró una excelente robustez y precisión frente a la mayoría de los ataques geométricos, espaciales y temporales, además de tener una muy buena eficiencia computacional. El rendimiento se evaluó mediante la precisión media promedio (mAP). De los resultados obtenidos se concluye que el algoritmo propuesto es adecuado para la indexación y recuperación de vídeos casi idénticos o la detección de manipulaciones.

Agradecimientos

Quiero expresar mi profundo agradecimiento a mi madre y a mi hermana por su constante apoyo y motivación durante el proceso de investigación y desarrollo de este trabajo. A pesar de no tener un conocimiento técnico en esta área de investigación, han sido incansables y han estado a

mi lado en todo momento. Su presencia incondicional ha sido esencial para mí y les estoy eternamente agradecido por ello. Además, quiero agradecer también a mi tutor por su guía experta y valiosa orientación a lo largo de todo este proceso.

BIBLIOGRAFÍA

[1] Tang Z, Song J, Zhang, X.Q., Sun R (2016) Multiple-image encryption with bit-plane decomposition and chaotic maps. Optics and Lasers in Engineering. 80, 1-11

[2] Kordopatis-Zilos G, Papadopoulos S, Patras I, Kompatsiaris I (2019) FIVR: Fine-grained incident video retrieval. IEEE Transactions on Multimedia. 21(10), 2638- 2652

[3] Tang Z, Zhang H, Pun C.M., Yu M, Yu C, Zhang X (2020) Robust image hashing with visual attention model and invariant moments. IET Image Processing. 14(5), 901- 908

[4] Tang Z, Dai Y, Zhang X.Q., Huang L, Yang F (2014) Robust image hashing via colour vector angles and discrete wavelet transform. IET Image Processing. 8(3), 142-149

[5] Huang Z, Liu S (2021) Perceptual hashing with visual content understanding for reduced-reference screen content image quality assessment. IEEE Transactions on Circuits and Systems for Video Technology. 31(7), 2808- 2823

[6] Qin C, Liu E, Feng G, Zhang X.P. (2021) Perceptual image hashing for content authentication based on convolutional neural network with multiple constraints. IEEE Transactions on Circuits and Systems for Video Technology. DOI:10.1109/TCSVT.2020.3047142, 1-15 (In press)

[7] Coskun B, Sankur B, Memon N (2006) Spatio-temporal transform-based video hashing. IEEE Trans Multimed 8(6), 1190-1208

[8] Tang Z, Chen L, Zhang X.Q., Zhang S (2018) Robust image hashing with tensor decomposition. IEEE Transactions on Knowledge and Data Engineering. 31(3), 549- 560

[9] Tang Z, Huang Z, Zhang X.Q., Lao H (2017) Robust image hashing with multidimensional scaling. Signal Processing. 137, 240-250

[10] Sandeep R, Sharma S, Bora P.K. (2017) Perceptual video hashing using 3D-radial projection technique. In: The Fourth International Conference on Signal Processing, Communication and Networking. Chennai, India, pp. 1-6

[11] Law-To J et al. (2007) Video copy detection: a comparative study. Proceedings of the 6th ACM international conference on Image and video retrieval. ACM

[12] Shinde SR, and Chiddarwar GG (2015) Recent advances in content-based video copy detection. 2015 International Conference on Pervasive Computing (ICPC). IEEE

[13] Zhang Z, Cao C, Zhang R, Zou J (2010) Video copy detection based on speeded up robust features and locality sensitive hashing. In: 2010 IEEE international conference on automation and logistics (ICAL). IEEE, pp 13–18

[14] Lee S, Yoo CD (2008) Robust video fingerprinting for content-based video identification. IEEE Trans Circuits Syst Video Technol 18(7):983–988

[15] Hua XS, Chen X, Zhang HJ (2004) Robust video signature based on ordinal measure. In: 2004 International conference on image processing (ICIP), vol 1. IEEE, pp 685-688

[16] Su X, Huang T, Gao W (2009) Robust video fingerprinting based on visual attention regions. In: 2009 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, pp 1525-1528

[17] Zheng L, Qiu G, Huang J, Fu H (2011) Salient covariance for near-duplicate image and video detection. In: 2011 18th IEEE international conference on image processing (ICIP). IEEE, pp 2537–2540

[18] Sarkar A, Singh V, Ghosh P, Manjunath BS, Singh A (2010) Efficient and robust detection of duplicate videos in a large database. IEEE Trans Circuits Syst Video Technol 20(6):870–885

[19] Sun R, Yan X, Gao J (2017) Robust video fingerprinting scheme based on contourlet hidden Markov tree model. Opt Int J Light Electron Opt 128:139–147

[20] Gu X, Zhang D, Zhang Y, Li J, Zhang L (2013) A video copy detection algorithm combining local feature's robustness and global feature's speed. In: 2013 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, pp 1508–1512

[21] D. Lowe, "Distinctive image features from scale-invariant keypoints," Int Journal of Computer Vision, vol. 60, pp. 91-110, 2004.

[22] Chen L, Stentiford FW (2008) Video sequence matching based on temporal ordinal measurement. Pattern Recogn Lett 29(13):1824–1831

[23] Radhakrishnan R, Bauer C (2008) Robust video fingerprints based on subspace embedding. In: 2008 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, pp 2245–2248

[24] Tasdemir K, Cetin AE (2010) Motion vector-based features for content-based video copy detection. In: 2010 20th International conference on pattern recognition (ICPR). IEEE, pp 3134–3137

[25] Zhang JR et al. (2012) Fast near-duplicate video retrieval via motion time series matching. 2012 IEEE International Conference on Multimedia and Expo. IEEE

[26] Esmaeili MM, Fatourechi M, Ward RK (2011) A robust and fast video copy detection system using content-based fingerprinting. IEEE Trans Inf Forensics Secur 6(1):213–226

[27] Sun J, Liu X, Wan W, Li J, Zhao D, Zhang H (2016) Video hashing based on appearance and attention features fusion via DBN. Neurocomputing 213:84–94

[28] Nie X, Yin Y, Sun J, Liu J, Cui C (2017) Comprehensive feature-based robust video fingerprinting using tensor model. IEEE Trans Multimed 19(4):785–796

[29] Douze M, Jégou H, Schmid C (2010) An image-based approach to video copy detection with spatio-temporal post-filtering. IEEE Trans Multimed 12(4):257–266

[30] M. Heikkila, M. Pietikainen, and C. Schmid (2009) Description of interest regions with local binary patterns, Pattern Recognit., vol. 42, no. 3, pp. 425–436

[31] Kim S, Choi JY, Han S, Ro YM (2014) Adaptive weighted fusion with new spatial and temporal fingerprints for improved video copy detection. Sig Process Image Commun 29(7):788–806

[32] Rameshnath S, Bora P.K. (2019) Perceptual video hashing based on temporal wavelet transform and random projections with application to indexing and retrieval of near-identical videos. Multimed Tools Appl 78, 18055–18075 https://doi.org/10.1007/s11042-019-7189-0

[33] Hu Y, Lu X (2018) Learning spatial-temporal features for video copy detection by the combination of CNN and RNN. J Vis Commun Image Represent 55:21–29

[34] Li J, Zhang H, Wan W, Sun J (2018) Two-class 3D-CNN classifiers combination for video copy detection. Multimed Tools Appl. https://doi.org/10.1007/s11042-018-6047-9

[35] Anuranji R, and Srimathi H (2020) A supervised deep convolutional based bidirectional long short-term memory video hashing for large scale video retrieval applications. Digit. Signal Process. 102, 102729. https://doi.org/10.1016/j.dsp.2020.102729

[36] Nie X, Zhou X, Shi Y, Sun J, and Yin Y (2021) Classification-enhancement deep hashing for large-scale video retrieval. Appl. Soft Comput. 109, 107467. doi:10. 1016/j.asoc.2021.107467

[37] Mahanty A (2022) Near Duplicate Video Detection (Perceptual Video Hashing) https://doi.org/10.5281/zenodo.4448295

[38] Chen H, Wo Y, Han G (2018) Multi-granularity geometrically robust video hashing for tampering detection. Multimedia Tools and Applications. 77(5), 5303-5321

[39] ReefVid: Free Reef Video Clip Database. (2023) http://www.reefvid.org/ Accessed 7 Jan 2023

[40] Open Video: A shared digital video collection. (2023) https://open-video.org/ Accessed 7 Jan 2023