

UIT-R

Sector de Radiocomunicaciones de la UIT

Recomendación UIT-R BT.1907-0 (01/2012)

Técnicas de medición objetiva de la calidad de percepción de vídeo para las aplicaciones de radiodifusión que utilizan TVAD en presencia de una señal de referencia completa

Serie BT
Servicio de radiodifusión (televisión)



Unión
Internacional de
Telecomunicaciones

Prólogo

El Sector de Radiocomunicaciones tiene como cometido garantizar la utilización racional, equitativa, eficaz y económica del espectro de frecuencias radioeléctricas por todos los servicios de radiocomunicaciones, incluidos los servicios por satélite, y realizar, sin limitación de gamas de frecuencias, estudios que sirvan de base para la adopción de las Recomendaciones UIT-R.

Las Conferencias Mundiales y Regionales de Radiocomunicaciones y las Asambleas de Radiocomunicaciones, con la colaboración de las Comisiones de Estudio, cumplen las funciones reglamentarias y políticas del Sector de Radiocomunicaciones.

Política sobre Derechos de Propiedad Intelectual (IPR)

La política del UIT-R sobre Derechos de Propiedad Intelectual se describe en la Política Común de Patentes UIT-T/UIT-R/ISO/CEI a la que se hace referencia en el Anexo 1 a la Resolución UIT-R 1. Los formularios que deben utilizarse en la declaración sobre patentes y utilización de patentes por los titulares de las mismas figuran en la dirección web <http://www.itu.int/ITU-R/go/patents/es>, donde también aparecen las Directrices para la implementación de la Política Común de Patentes UIT-T/UIT-R/ISO/CEI y la base de datos sobre información de patentes del UIT-R sobre este asunto.

Series de las Recomendaciones UIT-R

(También disponible en línea en <http://www.itu.int/publ/R-REC/es>)

Series	Título
BO	Distribución por satélite
BR	Registro para producción, archivo y reproducción; películas en televisión
BS	Servicio de radiodifusión sonora
BT	Service de radiodifusión (televisión)
F	Servicio fijo
M	Servicios móviles, de radiodeterminación, de aficionados y otros servicios por satélite conexos
P	Propagación de las ondas radioeléctricas
RA	Radioastronomía
RS	Sistemas de detección a distancia
S	Servicio fijo por satélite
SA	Aplicaciones espaciales y meteorología
SF	Compartición de frecuencias y coordinación entre los sistemas del servicio fijo por satélite y del servicio fijo
SM	Gestión del espectro
SNG	Periodismo electrónico por satélite
TF	Emisiones de frecuencias patrón y señales horarias
V	Vocabulario y cuestiones afines

Nota: Esta Recomendación UIT-R fue aprobada en inglés conforme al procedimiento detallado en la Resolución UIT-R 1.

Publicación electrónica
Ginebra, 2020

© UIT 2020

Reservados todos los derechos. Ninguna parte de esta publicación puede reproducirse por ningún procedimiento sin previa autorización escrita por parte de la UIT.

RECOMENDACIÓN UIT-R BT.1907-0*

**Técnicas de medición objetiva de la calidad de percepción de vídeo
para las aplicaciones de radiodifusión que utilizan TVAD
en presencia de una señal de referencia completa**

(2012)

Cometido

En la presente Recomendación se especifican métodos para estimar la calidad percibida de vídeo para las aplicaciones de radiodifusión que utilizan TVAD en presencia de una señal de referencia completa.

La Asamblea de Radiocomunicaciones de la UIT,

considerando

- a) que la capacidad de medir automáticamente la calidad del vídeo de radiodifusión se ha considerado desde hace tiempo como un activo valioso para la industria;
- b) que la Recomendación UIT-R BT.1683 describe técnicas de medición objetiva de la calidad de percepción de vídeo para la radiodifusión de televisión digital de definición convencional en presencia de una señal de referencia completa;
- c) que la Recomendación UIT-R BT.709 describe valores de parámetro para las normas de TVAD para la producción y el intercambio internacional de programas y que la Recomendación UIT-R BT.500 describe métodos de evaluación subjetiva de la calidad de imagen incluida la televisión de alta definición;
- d) que la TVAD se utiliza cada vez más en la radiodifusión;
- e) que, basándose en los resultados del informe sobre TVAD enviado por el VQEG, la Comisión de Estudio 9 del UIT-T ha elaborado la Recomendación UIT-T J.341, que especifica la medición objetiva de la calidad de percepción de vídeo de la TVAD en presencia de una señal de referencia completa;
- f) que la medición objetiva de la calidad de percepción de vídeo de la TVAD podrá complementar métodos de evaluación subjetiva,

recomienda

que se utilice el modelo de medición objetiva de la calidad de vídeo que se presenta en los Anexos 1, 2 y 3 para la medición objetiva de la calidad de percepción de vídeo para las aplicaciones de radiodifusión que utilizan TVAD en presencia de una señal de referencia completa.

* La Comisión de Estudio 6 de Radiocomunicaciones introdujo modificaciones redaccionales en esta Recomendación en febrero de 2020, de conformidad con la Resolución UIT-R 1.

Anexo 1

1 Introducción

En la presente Recomendación se propone un método de medición de la calidad de percepción de vídeo para su uso en aplicaciones no interactivas de televisión de alta definición (TVAD) cuando sea posible utilizar el método de medición de referencia completa. El modelo se elaboró para la estimación de notas de calidad subjetivas.

El método de medición de referencia completa puede utilizarse cuando la señal de vídeo de referencia no degradada está fácilmente disponible en el punto de medida, como puede suceder con las mediciones realizadas en equipos individuales, en una cadena en el laboratorio o en un entorno cerrado, como una estación de radiodifusión de televisión. El método de estimación incluye la estimación de la calibración y de la calidad objetiva de vídeo.

El material para la prueba de validación contenía degradaciones de codificación H.264 y MPEG-2, y condiciones que reproducían errores de transmisión (por ejemplo, bits erróneos y paquetes descartados). El modelo que figura en la presente Recomendación puede utilizarse para supervisar la calidad en redes activas a fin de garantizar su disponibilidad operativa. Los efectos visuales de las degradaciones podrán incluir degradaciones espaciales y temporales. El modelo que figura en la presente Recomendación también se puede utilizar para realizar pruebas de laboratorio de sistemas de vídeo. Si se emplea para comparar sistemas de vídeo, se aconseja utilizar un método cuantitativo (como el de la Recomendación UIT-T J.149) para determinar la precisión del modelo en dicho contexto en concreto.

La presente Recomendación se considera apropiada para servicios de radiodifusión prestados a una velocidad de entre 1 Mbit/s y 30 Mbit/s. En las pruebas de validación se consideraron las siguientes resoluciones y velocidades de trama:

- 1080/59.94/I
- 1080/25/P
- 1080/50/I
- 1080/29.97/P.

Las pruebas de validación de cada resolución se realizaron en las condiciones siguientes:

Factores de la prueba
Resolución de vídeo: 1920x1080 entrelazada y progresiva
Velocidad de las tramas de vídeo: 29,97 y 25 tramas por segundo
Velocidades binarias de vídeo: 1 a 30 Mbit/s
Congelación temporal de las tramas (pausa con salto) de un máximo de 2 segundos
Errores de transmisión con pérdida de paquetes
Conversión del SRC de 1080 a 720/P, compresión, transmisión, descompresión y conversión de nuevo a 1080
Tecnologías de codificación
H.264/AVC (MPEG-4 Parte 10)
MPEG-2

Cabe señalar que, en el plan de pruebas de validación, se consideró 720/P como parte de las condiciones de prueba (HRC). Puesto que en la actualidad las pantallas suelen estar adaptadas a la resolución 720/P, se estimó que los circuitos HRC de 720/P serían más adecuados para este formato.

1.1 Aplicación

Las aplicaciones para el modelo de estimación descrito en la presente Recomendación, incluyen, entre otras:

- 1) la supervisión potencial de la calidad en servicio, en tiempo real, en la fuente;
- 2) la supervisión remota de la calidad de destino cuando hay disponible una copia de la fuente en el punto de medición;
- 3) la medición de la calidad para la supervisión de un sistema de almacenamiento o de transmisión que utilice técnicas de compresión y descompresión de vídeo, ya sea en un solo paso o mediante concatenación de dichas técnicas;
- 4) Prueba en laboratorio de sistemas de vídeo.

1.2 Limitaciones

El modelo de estimación de la calidad de vídeo que se describe en la presente Recomendación no puede utilizarse para sustituir las pruebas objetivas. Los valores de la correlación entre dos pruebas subjetivas cuidadosamente diseñadas y realizadas (es decir, en dos laboratorios diferentes) suelen estar comprendidos entre 0,95 y 0,98. Si la presente Recomendación se utiliza para comparar sistemas de vídeo (por ejemplo, para comparar dos códecs), se aconseja utilizar un método cuantitativo para determinar la precisión del modelo en ese contexto concreto.

En aquellos casos en los que hubo congelación de tramas, las condiciones de la prueba mostraron por lo general una congelación de la trama inferior a los 2 segundos. El modelo que figura en la presente Recomendación no se validó para medir la calidad de vídeo en una situación de realmacenamiento (es decir, vídeo con un retardo creciente de forma persistente o con congelación sin salto). El modelo no se probó con velocidades de trama distintas de las que emplean los sistemas de televisión (a saber, 29,97 tramas por segundo y 25 tramas por segundo, en modo entrelazado o progresivo).

En el caso de nuevas tecnologías de codificación y transmisión que introduzcan efectos secundarios que no hayan sido tenidos en cuenta en la presente evaluación, el modelo objetivo puede dar resultados erróneos. En ese caso, es necesaria una evaluación subjetiva.

2 Referencias

Ninguna.

3 Definiciones

3.1 Términos definidos en otros textos

En la presente Recomendación se utilizan los siguientes términos definidos en otros textos:

3.1.1 evaluación subjetiva (imagen): Determinación de la calidad o del deterioro de imágenes tipo programa presentadas a un grupo de evaluadores humanos en sesiones de observación visual.

3.1.2 medición perceptual objetiva (imagen): Medición de las características de una cadena de programa utilizando imágenes tipo programa y métodos de medición objetiva (con instrumentos) para obtener un indicio que se acerque a la nota que se obtendría con una prueba de evaluación subjetiva.

3.1.3 proponente: Organización o empresa que propone un modelo de calidad de vídeo para la prueba de validación y su posible inclusión en una Recomendación de la UIT.

3.2 Términos definidos en la presente Recomendación

En la presente Recomendación se definen los términos siguientes:

3.2.1 velocidad de trama: Número de tramas únicas (es decir, total de tramas – tramas repetidas) por segundo.

3.2.2 errores de transmisión simulados: Errores impuestos en el tren de bits del vídeo digital en un entorno altamente controlado. Entre los ejemplos de este tipo de errores se encuentran las tasas de pérdidas de paquetes simuladas y los errores en los bits simulados. Los parámetros que se emplean para controlar los errores de transmisión simulados están bien definidos.

3.2.3 errores de transmisión: Errores impuestos en la transmisión del vídeo. Entre los ejemplos de este tipo de errores se encuentran los errores de transmisión simulados y las condiciones de red en servicio.

4 Abreviaturas y siglas

En la presente Recomendación se emplean las abreviaturas y siglas siguientes:

ACR	Índice por categoría absoluta (véase UIT-R BT.500)
ACR-HR	Índice por categoría absoluta con referencia escondida (véase UIT-T P.910)
AVI	Entrelazado de audio vídeo
DMOS	Diferencia de la nota media de opinión
FR	Referencia completa
FRTV	Televisión de referencia completa
HRC	Circuito ficticio de referencia
ILG	Grupo de laboratorio independiente del VQEG
MOS	Nota media de opinión
MOSp	Nota media de opinión, predicción
NR	Sin referencia (o referencia cero)
PSNR	Relación señal de cresta/ruido
PVS	Secuencia de vídeo procesada
RMSE	Error cuadrático medio
RR	Referencia reducida
SFR	Velocidad de trama de fuente
SRC	Canal o circuito de referencia de fuente
VQEG	Grupo de Expertos en calidad del vídeo
YUV	Espacio de color y formato de archivo

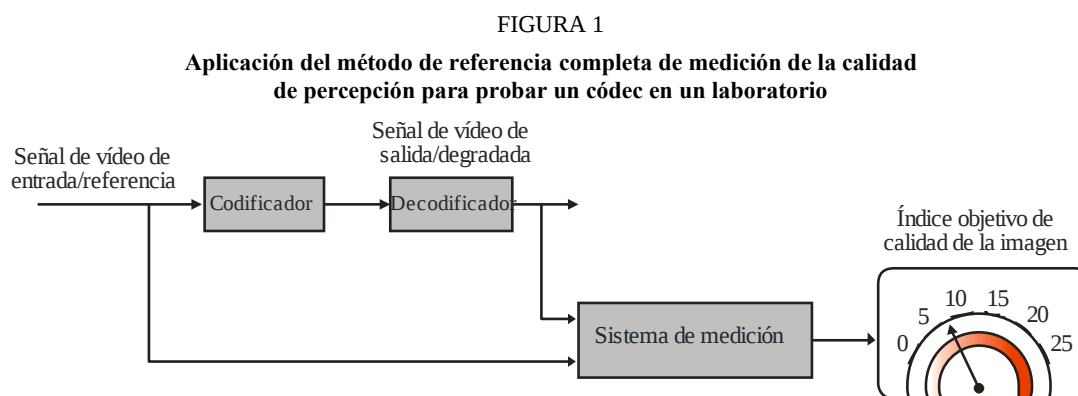
5 Convenciones

Ninguna.

6 Descripción de la metodología de referencia completa

El método de medición de doble extremo con referencia completa para la medición objetiva de la calidad de percepción de vídeo evalúa el comportamiento de los sistemas comparando la señal de vídeo de entrada, o de referencia, sin distorsionar a la entrada del sistema con la señal degradada a la salida del sistema (Fig. 1).

En la Figura 1 se muestra un ejemplo de aplicación del método de referencia completa para probar un códec en el laboratorio.



La comparación entre las señales de entrada y salida puede requerir un procedimiento de alineamiento temporal o espacial; el segundo tiene por objeto compensar los posibles recortes o desplazamientos en sentido horizontal o vertical de la imagen. También puede requerir que se corrijan las diferencias de traslación o de ganancia, tanto en los canales de luminancia como en los de crominancia. A continuación se calcula el índice objetivo de calidad de imagen, normalmente aplicando un modelo de percepción de visión humana.

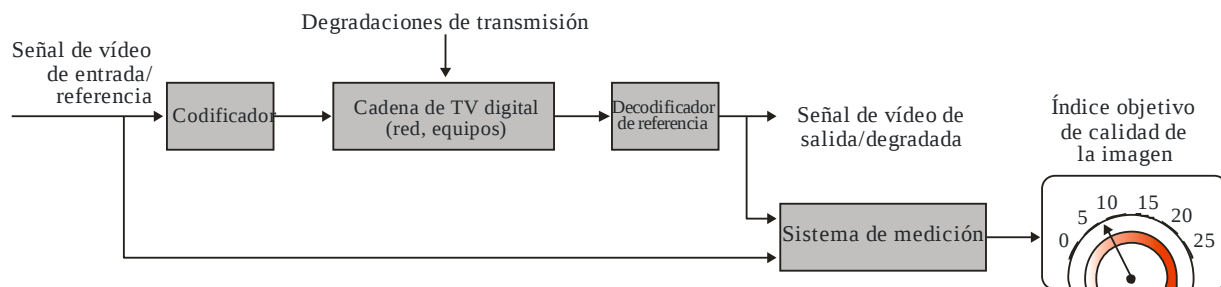
Los ajustes de alineamiento y ganancia se conocen como registro. Este proceso es necesario porque la mayoría de métodos de referencia completa comparan píxel a píxel las características extraídas de las imágenes de referencia y las imágenes procesadas. Las métricas de la calidad de vídeo descritas en el Anexo 2 incluyen métodos de registro.

Como las métricas de la calidad de vídeo se basan normalmente más en aproximaciones a respuestas visuales humanas que en mediciones de fenómenos de codificación específicos, en principio son igualmente válidas para sistemas analógicos o digitales. En principio, también son válidas para cadenas que combinan sistemas analógicos y digitales o en las que se concatenan sistemas de compresión digital.

La Fig. 2 muestra un ejemplo de aplicación del método de referencia completa para probar una cadena de transmisión.

FIGURA 2

Aplicación del método de referencia completa de medición de la calidad de percepción para probar una cadena de transmisión



BT.1907-02

En este caso, el decodificador de referencia se alimenta desde varios puntos de la cadena de transmisión; por ejemplo, el decodificador puede estar situado en un punto de la red, como en la Fig. 2, o directamente en la salida del codificador, como en la Fig. 1. Si la cadena de transmisión digital es transparente, la medición del índice objetivo de calidad de la imagen en la fuente es igual a la medición en cualquier otro punto posterior de la cadena.

Generalmente, se acepta que el método de referencia completa es el más preciso para medir la calidad de percepción de la imagen. Se ha demostrado que el método tiene potencial para una alta correlación con las evaluaciones subjetivas realizadas de conformidad con los métodos ACR-HR especificados en la Recomendación UIT-T P.910.

7 Conclusiones del Grupo de Expertos en calidad del vídeo (VQEG)

Los estudios de las mediciones de la calidad de percepción de vídeo se han llevado a cabo en el seno de un grupo informal, denominado Grupo de Expertos en calidad del vídeo (VQGE), que informa de sus resultados a las Comisiones de Estudio 9 y 12 del UIT-T y a la Comisión de Estudio 6 del UIT-R. La prueba de fase I sobre televisión de alta definición recientemente completada por el VQEG evaluó el comportamiento de los algoritmos de referencia completa propuestos para medir la calidad de percepción de vídeo.

Las estadísticas que se ofrecen a continuación proceden del informe final del VQEG sobre TVAD. Cabe señalar que el cuerpo del informe del VQEG sobre TVAD incluye otras métricas, entre ellas la Correlación de Pearson y el RMSE calculado en experimentos individuales, intervalos de confianza, pruebas de significación estadística realizadas en experimentos individuales, análisis de los subconjuntos de datos que incluyen degradaciones concretas (por ejemplo, solamente para la codificación H.264), diagramas de dispersión y los coeficientes para los que los valores son válidos.

Análisis primario

El comportamiento del modelo de referencia completa se resume en el Cuadro 1. La PSNR se calcula según lo establecido en la Recomendación UIT-T J.340 y se incluye en este análisis con fines comparativos. «El RMSE del superconjunto» identifica la métrica primaria (RMSE) calculada en el superconjunto combinado (es decir, los seis experimentos organizados en una única escala). «Total de grupos de mejor comportamiento» identifica el número de experimentos (0 a 6) en los que este modelo fue el modelo de mejor comportamiento o su equivalente estadístico. «Mejor que la PSNR total» identifica el número de experimentos (0 a 6) en los que el modelo fue mejor que la PSNR. «Mejor que la PSNR del superconjunto» permite comprobar si cada modelo es estadísticamente mejor que la PSNR en el superconjunto combinado. «Correlación del superconjunto» identifica la Correlación de Pearson calculada en el superconjunto combinado.

CUADRO 1

Métrica	PSNR	SwissQual
RMSE del superconjunto	0,71	0,56
Total de grupos de mejor comportamiento	1	5
Mejor que la PSNR total	–	4
Mejor que la PSNR del superconjunto	–	Sí
Correlación del superconjunto	0,78	0,87

Anexo 2

Descripción del modelo

Nota del Editor: El código fuente que debe incorporarse a esta sección conforma la parte obligatoria de la presente Recomendación y está disponible en <http://ifatemp.itu.int/t/2009/sg9/exchange/q2/>.

Descripción general del modelo

El modelo predice la calidad de vídeo tal y como la perciben los participantes en un experimento. El modelo de predicción emplea modelos de inspiración cognitiva y psicovisual para emular la percepción subjetiva.

Al tratarse de un enfoque de referencia completa, el modelo compara la señal de vídeo de entrada o de referencia de alta calidad con la secuencia de vídeo degradada conexas examinada. Este proceso se muestra en la Fig. 3.

La estimación del resultado se basa en los pasos siguientes:

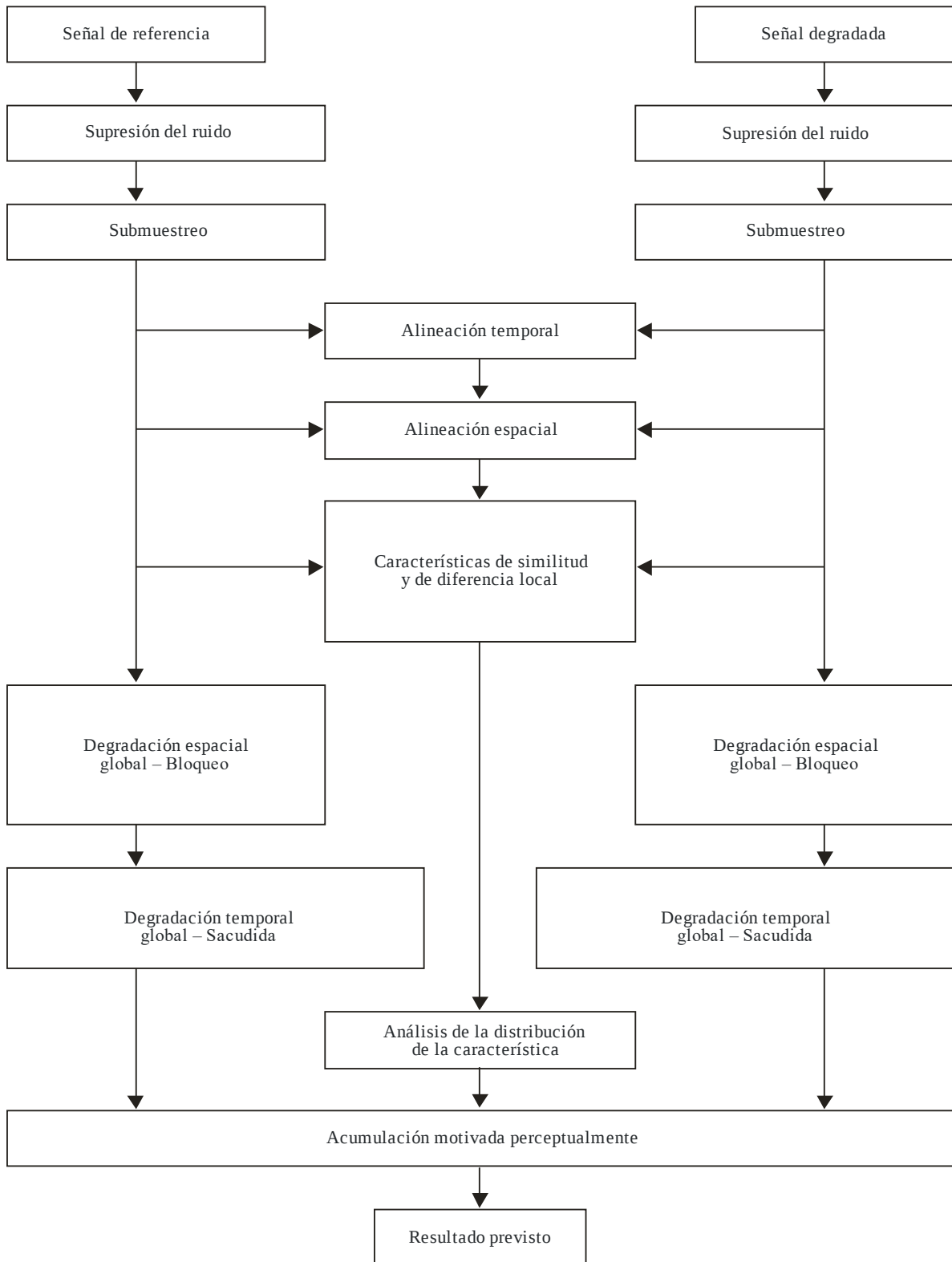
- 1) En primer lugar, se preprocesan las secuencias de vídeo. En especial, se filtran las tramas para eliminar el ruido y se hace un submuestreo de las tramas.
- 2) Se efectúa una alineación temporal de las tramas entre la secuencia de vídeo de referencia y la procesada.
- 3) Se efectúa una alineación espacial de la trama de vídeo procesada y la trama de vídeo de referencia correspondiente.
- 4) Se calculan las características de la calidad espacial local, una medida de la similitud y de la diferencia local inspiradas por la percepción visual.
- 5) Se efectúa un análisis de la distribución de la similitud y de la diferencia local.
- 6) Se mide la degradación espacial global utilizando una característica de *bloqueo*.
- 7) Se mide la degradación temporal global utilizando una característica de *sacudida*. La medición de la sacudida se calcula evaluando la intensidad de movimiento global y local y el tiempo de exhibición de la trama.

- 8) Se estima la nota de calidad a partir de la suma no lineal de las características anteriores.
- 9) Para evitar errores de predicción si existiera un alineamiento espacial incorrecto relativamente alto entre la secuencia de vídeo procesada y la secuencia de vídeo de referencia, los pasos anteriormente mencionados se calculan para 3 alineamientos espaciales horizontales y verticales distintos de la secuencia de vídeo y se adopta, como valor final de la calidad estimada, el valor máximo previsto de entre todas las posiciones espaciales.

Los pasos individuales se explican con más detalle entre los § 2.1 y 2.9. El § 2.10 contiene un archivo integrado con código fuente en lenguaje C++ que abarca las partes y las funciones esenciales para una aplicación acorde con la descripción del modelo. Los nombres de las funciones en lenguaje C++ que se mencionan entre los § 2.1 y 2.9 se refieren a este código fuente de referencia (por ejemplo, § 2.2 se refiere a `CFrameAnalysisFullRef::ContentTimeAlignment`).

FIGURA 3

Descripción general de un diagrama de flujo de las etapas de procesamiento del modelo. En la parte superior, la entrada corresponde a la secuencia de vídeo de referencia y a la secuencia de vídeo degradada (o procesada). Los distintos pasos del procesamiento generan el resultado del modelo principal, y el resultado previsto se da en la parte inferior



2.1 Preprocesamiento

Cada trama de las secuencias de vídeo de referencia y procesada se filtra espacialmente a través de un filtro paso bajo y se submuestra en tres resoluciones distintas, R1, R2, R3:

	trama original		R1		R2		R3
altura × anchura	1080 × 1920	→	540 × 960	→	270 × 480	→	96 × 128

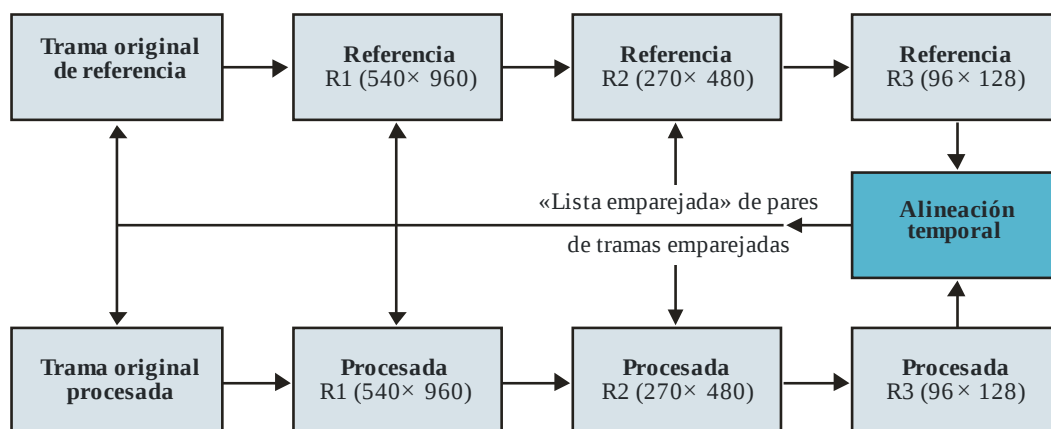
Véase el método `CFrameAnalysisFullRef::ContentTimeAlignment` para generar tramas en la resolución R3 y `CFrameSeq::ReadFrame` para generar tramas en las resoluciones R1 y R2.

Cabe señalar que la aplicación no es directa a causa de limitaciones de memoria.

En la Figura 4 se muestran las tres resoluciones submuestreadas.

FIGURA 4

Las tramas de la secuencia de vídeo de referencia y procesada se filtran mediante un filtro paso bajo y se muestrean en tres resoluciones distintas. La resolución más pequeña R3 se utiliza para llevar a cabo la alineación temporal de la trama. La lista resultante de tramas emparejadas puede utilizarse para emparejar las tramas en cualquier otra resolución



BT.1907-04

2.2 Alineación temporal

Para la alineación temporal, se emplean las secuencias de vídeo de referencia y procesada en la resolución baja R3.

La alineación temporal se efectúa de manera recurrente del modo siguiente:

- 1) Se busca una «trama de guía» en la secuencia de referencia (`Ref_anchor`).
- 2) Se empareja esta trama con la trama con mejor concordancia de la secuencia degradada (`Deg_best_match`).

Se toma la trama con mejor concordancia de la secuencia degradada (Deg_best_match) y se empareja con tramas próximas a la «trama de guía» de la secuencia de referencia (Ref_anchor). A partir de un criterio de similitud, se intenta encontrar una concordancia mejor entre Deg_best_match y las tramas del entorno de Ref_anchor, que se almacena como par con mejor concordancia. Como criterio de similitud entre el plano Y de la trama procesada x y la trama de referencia y , se utiliza la función:

$$\text{sim} = \exp(-\text{mean_square_diff}(a*x+b, y)) \quad (2.1)$$

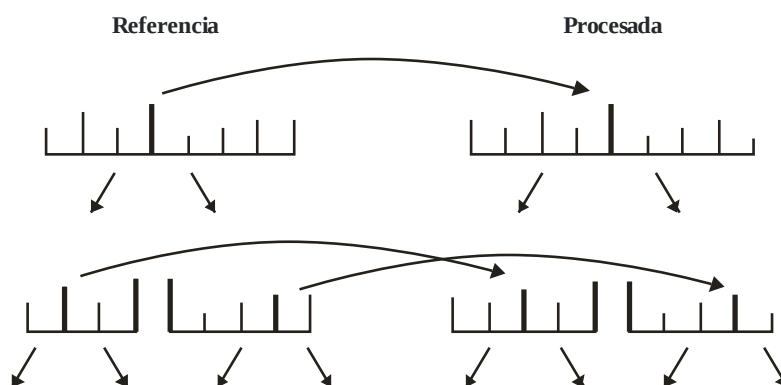
con unos parámetros a, b que minimizan la diferencia cuadrática media entre los valores del plano Y de la trama procesada x y la trama de referencia y , véase el método

FrameSimilarity::similarity en la aplicación de referencia.

- 1) Si la concordancia entre el par de tramas que concuerdan es buena (el criterio de similitud está por encima de un umbral de aceptación), se divide la secuencia de vídeo de referencia y la secuencia de vídeo procesada del par de tramas que concuerdan en dos secuencias de vídeo, antes y después de las tramas que concuerdan. Se empieza en 1) con ambos pares de subsecuencias de referencia y degradada.
- 2) Si la concordancia entre el par de tramas que concuerdan no es «buena», se empieza de nuevo en 1) con una «trama de guía» distinta del vídeo de referencia. Dado que no existe un conocimiento *a priori* del valor esperado para una trama con «buena» concordancia, el umbral de concordancia se reduce iterativamente. Los valores siguientes se determinaron a partir de distintas muestras de datos de prueba: el umbral inicial con respecto al criterio de similitud de la ecuación (2.1) es 0,98. Al no lograr ninguna concordancia con 10 tramas de guía, se reduce en 0,98, y se reinicia la concordancia en 1). De este modo, al menos se intenta lograr una concordancia con otras 10 tramas de guía como máximo; en caso de fracasar, se vuelve a reducir el límite. Se procede así hasta que se alcanza un valor mínimo de 0,1. Véase SQ_TimeAlignement::findAnchorAndDescend para todos los detalles relativos a la aplicación.

FIGURA 5

Ilustración del enfoque recurrente que se emplea en la alineación temporal. Se empareja una trama de guía de la secuencia de referencia con una trama de la secuencia procesada. A continuación, se dividen ambas secuencias y, en cada subsecuencia, se elige una trama de guía y se empareja



BT.1907-05

El resultado de la alineación temporal es una secuencia (fundamentalmente, una «lista de emparejamientos») en la que se asigna a cada trama de la secuencia de vídeo procesada una trama de la secuencia de referencia, o indicador, que indica que no pudieron encontrarse tramas con una

concordancia suficientemente buena. Así, en las últimas etapas del procesamiento, cada trama emparejada de la secuencia de vídeo procesada tiene una trama correspondiente del vídeo de referencia. Las tramas de la secuencia de vídeo procesada cuyo indicador reza «no emparejada» se compararán con las dos tramas de referencia que concuerdan con la trama «emparejada» anterior y posterior de la secuencia de vídeo procesada. Se ha optado por fijar un «límite de emparejamiento» muy bajo para que solamente las tramas muy degradadas tengan un indicador «no emparejada».

Véase el método `CFrameAnalysisFullRef::sqVTA_ContentFrameTimeAlignement_M` para todos los detalles relativos a la aplicación.

2.3 Alineación de la trama espacial

Se repite en todas las tramas de la secuencia de vídeo procesada y:

- 1) Si no existe concordancia para esta trama, se usa la alineación espacial anterior. Si existe concordancia para esta trama, se efectúa una alineación espacial entre la trama procesada y la trama de referencia correspondiente, según la lista de concordancia de la alineación temporal:
 - a) Para la primera trama, el valor inicial del desplazamiento espacial es de 0 (tanto en la dirección horizontal como en la vertical). Para las tramas posteriores, se usa la alineación espacial de la trama emparejada anterior.
 - b) Se repite en todos los desplazamientos espaciales posibles (horizontales y verticales), utilizando el límite del punto 2) *infra*. Si un desplazamiento espacial distinto provoca una diferencia significativamente menor (con respecto a la función de coste) entre la trama procesada y la trama de referencia correspondiente, se ajusta el desplazamiento espacial. Para la función de coste, se utiliza la función:

$$\text{rmse}(Y(dv, dh), Y_{\text{ref}}) + \text{abs}(dv) + \text{abs}(dh),$$
 donde Y denota el plano Y de la trama procesada en la resolución $R1$ e Y_{ref} denota la trama de referencia en la resolución $R1$, $Y(dv, dh)$ denota la trama desplazada Y , desplazada en dv y dh , siendo dv y dh los desplazamientos vertical y horizontal. En la función de coste se incluyen el segundo y el tercer término para favorecer pequeños desplazamientos espaciales. Nótese que, para el cálculo del error cuadrático medio, se desestima una pequeña línea de demarcación de las tramas para evitar un manejo más complicado de las líneas de demarcación.
 - c) De este modo, es posible compensar los desplazamientos espaciales dependientes del tiempo. Los errores de alineación en una trama se pueden corregir mediante la alineación de las tramas posteriores.
- 2) Este primer paso de la alineación automatizada del desplazamiento espacial se limita a ± 4 píxeles. Para desplazamientos espaciales mayores, véase § 2.9.
- 3) Después de la alineación espacial, cada trama de la secuencia de vídeo procesada tiene una trama de referencia correspondiente (o dos en el caso de tramas no emparejadas), según la lista de correspondencia de la alineación temporal y un factor de corrección bien definido del desplazamiento espacial. Esto permite comparar con precisión las tramas de la secuencia de vídeo procesada con las tramas de la referencia, aspecto fundamental para extraer las características siguientes.

Véase el método `CFrameAnalysisFullRef::DetermineSpatialAlignment` para todos los detalles relativos a la aplicación. El umbral constante del paso 2 (± 4 píxeles) puede incrementarse para aceptar desplazamientos espaciales mayores.

2.4 Cálculo de las características de similitud y diferencia local

Para cada par de tramas alineadas, se calcula un conjunto de características de la calidad espacial:

En primer lugar, se calcula una medida de la *similitud* y de la *diferencia* local mediante la iteración en las regiones cuadradas adyacentes y distribuidas de manera uniforme de tamaño 13×13 de la trama procesada y de la trama de referencia en la resolución R2. Como la resolución R2 no es divisible por 13, se ignora una pequeña línea de demarcación.

Las regiones locales reciben el nombre de s_prc y s_ref , y la similitud s y la diferencia D se calculan del modo siguiente:

$$S = (\text{cor}(s_prc, s_ref) + 25) / (\text{var}(s_ref) + 25) \quad (4.1)$$

$$D = \sqrt{\text{avg}((S * (s_prc - \text{mean}(s_prc)) - (s_ref - \text{mean}(s_ref)))^2))} \quad (4.2)$$

siendo cor la correlación y var la varianza de los valores del píxel en la región cuadrada correspondiente. La función avg calcula la media basándose en todos los píxeles de la región cuadrada, y sqrt denota la raíz cuadrada. Los valores D y S son los contribuyentes principales al valor de la calidad espacial.

En este punto, las características de similitud y diferencia, S y D , son una matriz de valores para cada trama, en la que un valor corresponde a cada región local cuadrada. A los fines de la calidad percibida, no solo es importante el valor medio, sino también la forma de la distribución de S y D , respectivamente.

2.5 Análisis de la distribución de las características locales

Esta sección empieza con la introducción de algunas anotaciones:

$\text{quantile}(X, c)$ denota el *cuantil* c de la distribución de los valores (entradas) de un vector o matriz X . Más concretamente, para un vector X y una constante c con $0 \leq c \leq 1$, el cuantil

$$q = \text{quantile}(X, c)$$

es el valor q , de modo que una fracción c de todos los valores de X es menor o igual a q .

La función *trimmed_mean* se define de la manera siguiente. Esta anotación se utilizará posteriormente. Para una matriz x , la media ajustada

$$\text{trimmedMean}(X, c)$$

es la media de todas las entradas de x entre los cuantiles c y $(1-c)$ de x .

Por ejemplo, $\text{trimmedMean}(X, 0.1)$ es la media de todos los valores de x después de ignorar el 10% de los valores más pequeños y el 10% de los valores más grandes de x .

La notación $X(X > c)$ denota el conjunto de todos los valores de x mayores de c . Por ejemplo,

$$\text{trimmedMean}(X, c) = \text{mean}(X(X > \text{quantile}(X, c) \text{ and } X < \text{quantile}(X, 1-c)))$$

Utilizando estas notaciones, se calculan los siguientes valores de las características basándose en s de la ecuación (4.1), y en D de la ecuación (4.2):

$$s_m = \text{trimmedMean}(S, c) \quad (5.1)$$

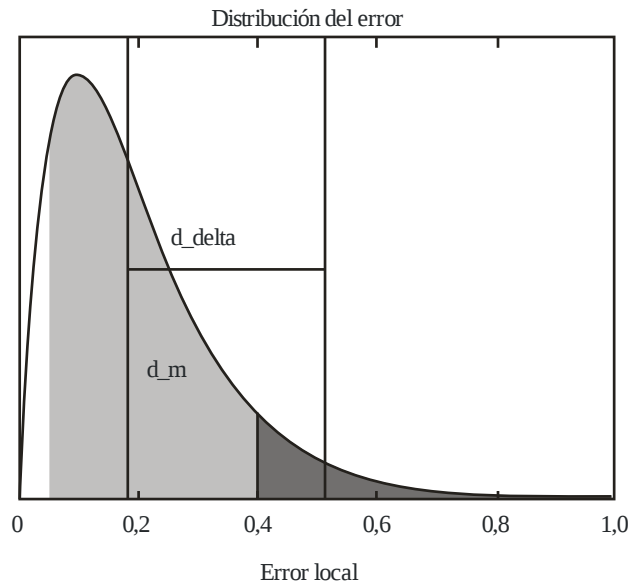
$$d_m = \text{trimmedMean}(D, c) \quad (5.2)$$

$$s_delta = s_m - \text{mean}(S(S < \text{quantile}(S, c))) \quad (5.3)$$

$$d_delta = \text{mean}(D(D > \text{quantile}(D, 1-c))) - d_m \quad (5.4)$$

con $c=0,2$. Este proceso se muestra en la Fig. 6, que presenta d_m y d_{Δ} .

FIGURA 6
Distribución de una característica local D. La media ajustada d_m corresponde a la media de la zona de color gris claro (línea vertical negra). El valor d_{Δ} corresponde a la diferencia de la media de los valores en las zonas de color gris oscuro y gris claro (línea horizontal)



BT.1907-06

Véase el método `CFrameAnalysisFullRef::ComputeSimilarity` para el cálculo de s y D .

2.6 Cálculo de la característica de bloqueo

La característica de *bloqueo* se calcula utilizando las tramas en la resolución R1. Esta característica mide la visibilidad de las líneas de demarcación del bloque debidas a errores de codificación y/o transmisión. A causa del cálculo en la resolución R1, automáticamente se centra la atención en la visibilidad de percepción de los límites. A partir de una descripción general, la característica de bloqueo calcula:

- 1) Las derivadas direccionales (imágenes de bordes) para los bordes horizontales y verticales. El resultado son dos matrices, una correspondiente a los bordes horizontales y otra a los bordes verticales, por cada trama de secuencia de vídeo, que reciben el nombre de `verGrad_n` y `horGrad_n` en el pseudocódigo *infra*.
- 2) Una suma por filas y por columnas del logaritmo de los bordes horizontales y verticales, que da como resultado dos vectores, uno de ellos correspondiente a la suma de los bordes horizontales y otro a la de los verticales, que reciben el nombre de `sumW` y `sumH` *infra*.
- 3) Una media de una submuestra de `sumW` y `sumH`, respectivamente, en un tamaño de paso n y con una desviación m , que se calcula mediante la función `vq_AvgSubsample` *infra*.

La idea es que una estructura sólida de bloques de tamaño n mostrará como diferencia importante `delta_edge` en `vq_AvgSubsample` en el tamaño de paso n calculado para distintas desviaciones.

Por ejemplo, una estructura de bloques de tamaño 4 en la trama original tiene una estructura de bloques de tamaño 2 en la resolución R1. En consecuencia, el cálculo de `vq_AvgSubsample(x, 2, 0)` y `vq_AvgSubsample(x, 2, 1)` debería mostrar una diferencia importante

en presencia de una estructura sólida de bloques. Para evitar una dependencia con respecto al contenido, los experimentos en los que se empleó una muestra amplia de secuencias de vídeo permitieron saber cómo relacionar la diferencia calculada medida en la secuencia de vídeo procesada con los valores de la secuencia de referencia.

El siguiente pseudocódigo explica mejor el cálculo. Aquí, `horGrad` y `verGrad` son las derivadas espaciales horizontales y verticales de una trama, obtenidas a partir de la diferencia de píxeles adyacentes,

$$\begin{aligned}\text{verGrad}_n(i,j) &= Y_n(i+1,j) - Y_n(i,j), y \\ \text{horGrad}_n(i,j) &= Y_n(i,j+1) - Y_n(i,j),\end{aligned}$$

siendo $Y_n(i,j)$ el valor del pixel en la posición (i,j) del plano Y de la trama n. La función

`vq_AvgSubsample( x, step, offset )`

calcula el valor medio del vector x en todas las muestras en un tamaño de paso `step` y a partir de una desviación `offset`.

```
// loop over all frames and compute:
for( UINT i=0; i<horGrad.Height; i++ ){
    for( UINT j=0; j<horGrad.Width; j++ ){
        w = (double)verGrad(i,j);
        h = (double)horGrad(i,j);
        // sum edges (-2: small differences can be the result of integer
        // values used to store frames)
        sumW(i) += log(1.0+max(0.0,fabs(w)-2.0));
        sumH(j) += log(1.0+max(0.0,fabs(h)-2.0));
    }
}

double dH0 = vq_AvgSubsample( sumH, 2, 0 );
double dH1 = vq_AvgSubsample( sumH, 2, 1 );
double dW0 = vq_AvgSubsample( sumW, 2, 0 );
double dW1 = vq_AvgSubsample( sumW, 2, 1 );

edge_max = 0.5 * (vq_Max(dW0,dW1) + vq_Max(dH0,dH1) );
edge_min = 0.5 * (vq_Min(dW0,dW1) + vq_Min(dH0,dH1) );

// now: denote by edge_max(i) the value of edge_max above, corresponding to
// frame i of the processed video sequence, and by edge_max_ref(i) the values
// edge_max above corresponding to frame i of the reference video sequence,
// and analogously for edge_min(i), edge_min_ref(i). Then compute:

for( UINT i=0; i<nbOfFramesInProcessedVideo; i++ ){
    // get frame nb of ref frame (according to match-list)
    UINT i_ref = (UINT)floor(ref_frameNb_all(i)+0.5f);

    float delta_edge = edge_max(i) - edge_min(i);
    float delta_edge_ref = edge_max_ref(i_ref) - edge_min_ref(i_ref);

    x(i) = vq_Max(0.0f,delta_edge - delta_edge_ref) / (1.0f+edge_max(i));
}
// blockiness(i) is then a non-linear monotone transform of x(i) ...
```

Nótese que, puesto que cabe la posibilidad de que haya tramas de 720 ampliadas en el muestreo, el cálculo es ligeramente más complicado:

Véase `vquad_hd::vq_BlockinessPhaseDiff` y `CQualityModelFullRef::Blockiness` para todos los detalles de aplicación relativos al cálculo de la característica de bloqueo.

2.7 Cálculo de la característica de sacudida (cualidad temporal)

La característica de *sacudida* se calcula buscando la media del producto del tiempo de exhibición relativo, la transformación no lineal del tiempo de exhibición y la transformación no lineal de la intensidad del movimiento. La intensidad del movimiento se obtiene principalmente a partir de las diferencias entre tramas en distintas regiones de la trama. El *tiempo de exhibición* es el tiempo, en milisegundos, en que una trama aparece en pantalla. Para determinar el tiempo de exhibición de cada trama, se efectúa un análisis de la intensidad del movimiento local, ya que las tramas en la secuencia de vídeo procesada podrían ser repeticiones de tramas anteriores.

La sacudida tiene en cuenta la cantidad de información que se pierde durante la reproducción de la secuencia de vídeo procesada. Cuando la secuencia se reproduce de manera fluida, su valor es muy bajo, mientras que aumenta si hay pausas o una velocidad de trama menor. Por otro lado, en el caso de un error de emparejamiento temporal fijo, la medición de la sacudida adopta unos valores mayores para secuencias de vídeo con una mayor intensidad de movimiento.

El pseudocódigo siguiente muestra los detalles. Las entradas corresponden al vector de intensidad del movimiento `motionInt`, al vector de probabilidad de repetición de la trama `repFrame` y a un vector de tiempo de exhibición de la trama `displayTime`. La salida es el vector `jerkiness`, que denota la sacudida en cada trama de la secuencia de vídeo procesada. Más concretamente, el vector `motionInt` denota el valor cuadrático medio de la diferencia entre tramas, medido en el plano Y en la resolución R2. El vector `repFrame` denota la probabilidad de repetición de trama, es decir que, en función de la intensidad del movimiento, cabe la probabilidad de que cada trama sea una repetición de la trama anterior: si la repetición con respecto a la trama anterior es perfecta, la probabilidad de que la trama real sea una repetición es de 1. Cuando la intensidad del movimiento es elevada, la probabilidad de que la trama real sea una repetición de la anterior es de 0. La probabilidad puede tener valores intermedios si la intensidad del movimiento tiene un valor muy pequeño pero distinto de 0. Explícitamente,

$$\text{repFrame}(i) = \begin{cases} 0 & \text{if } m(i) < p/2 \\ (m(i) - p/2) / p & \text{if } p/2 \leq m(i) < 3/2 * p \\ 1 & \text{if } 3/2 * p \leq m(i) \end{cases}$$

siendo $m(i)$ la intensidad de movimiento de la trama i . Para la prueba se eligió el parámetro $p=0,01$.

```
int vq_CalcJerkiness( const CVector<float> & motionInt,
                    const CVector<float> & repFrame,
                    const CVector<float> & displayTime,
                    CVector<float> & jerkiness ){

    // -----
    // the 4 parameters of the jerkiness measure: determined using a
    // large sample of video sequences containing temporal degradations
    // only.
    float a = 0.9f;
    float b = 5.0f;

    float aT = 40.0f;
    float bT = 5.0f;
```

```

// -----

// get probability of new frame = 1 - prob of repeated frame:
CVector<float> newFrame = repFrame*(-1.0f) + 1.0f;

// count number of non-repeated frames
float fNbRepeated = repFrame.Sum();
float fNbNonRepeated = repFrame.Length() - fNbRepeated;

// calculate jerkiness
float fR = 0.0f;
// look for frame repetition intervals (~ display time)
// of length i
for( UINT i=1; i<= iNbFrames; i++ )
{
    // look for frame repetitions starting at position j
    for( UINT j=0; j<iNbFrames-i+1; j++ )
    {
        float fP = newFrame(j);    // prob. : start of repetition block

        for( UINT k=1; k<i; k++ )
        {
            fP *= repFrame(j+k);    // prob. : all repeated frames
        }
        if( i+j < iNbFrames )
        {
            fP *= newFrame(j+i);    // prob. : end of repetition block
        }

        // calculate the display time (in s) of frame j,
        // if displayed from
        // time t_j until t_(j+i), which occurs with probability fP
        float fDispTime = displayTime.SumPart(j,j+i)/1000.0f;

        // -> measure jerking and add to result
        float fIFDiff = motionInt( j+i-1 );

        // normalisation values: such that at 0 jerkiness value is 0,
        // and saturates at 1
        float c = 1.0f / (1.0f+exp(b));
        float cT = 1.0f / (1.0f+exp(bT));

        float fJ = 1.0f / (1.0f + exp( -( a * fIFDiff - b ) ));
        float fJT = 1.0f / (1.0f + exp( -( aT * fDispTime - bT)));
        fJ = (fJ - c)/(1.0f - c);
        fJT = (fJT - cT)/(1.0f - cT);

        // jerkiness value: propability * interframeDiffFactor *
        // displayTimeFactor

        fR = fP * fJ * fJT * fDispTime;

        // add to jerkiness vector at position j+i (corresponding to the
        // end of display time)
        jerkiness( vq_Min(j+i,iNbFrames-1) ) += fR;
    }
}
return 0;
}

```

Véase el método `vquad_hd::vq_ProbOfRepeatedFrame` y `vquad_hd::vq_CalcJerkiness` para los detalles relativos a la aplicación.

2.8 Agregación a MOS

Las características anteriormente descritas, la similitud debida a s_m y s_{Δ} , la diferencia debida a d_m y d_{Δ} , $blockiness$ y $jerkiness$, son la base para la estimación del resultado, junto con el vector de tiempo de exhibición de la trama `displayTime`. Estos vectores tienen una duración igual al número de tramas de la secuencia de vídeo procesada.

Para trasladar estas características a una escala perceptual, se emplea una función parametrizada con forma de S:

$$S: \mathbb{R} \rightarrow \mathbb{R}, \quad y = S(x).$$

La función S se parametriza mediante tres parámetros (p_x, p_y, q) , cuya interpretación es la siguiente: (p_x, p_y) es la ubicación en $\mathbb{R} \times \mathbb{R}$ y q la pendiente del punto de inflexión, más concretamente:

$$S(x) = \begin{cases} a * x^b & \text{if } x \leq p_x \\ d / (1 + \exp(-c * (x - p_x))) + 1 - d & \text{else} \end{cases} \quad (8.1)$$

siendo:

$$\begin{aligned} a &= p_y / p_x^{(q * p_x / p_y)} \\ b &= q * p_x / p_y \\ c &= 4 * q / d \\ d &= 2 * (1 - p_y) \end{aligned}$$

En la Fig. 7 se muestra un gráfico de una función con forma de S con distintos parámetros. La función con forma de S empieza en el origen, crece de manera polinomial hasta alcanzar el punto de inflexión y satura exponencialmente al aproximarse a 1.

FIGURA 7
Funciones con forma de S, parametrizadas según la posición y la pendiente del punto de inflexión. Se muestran dos funciones de muestra para distintos parámetros

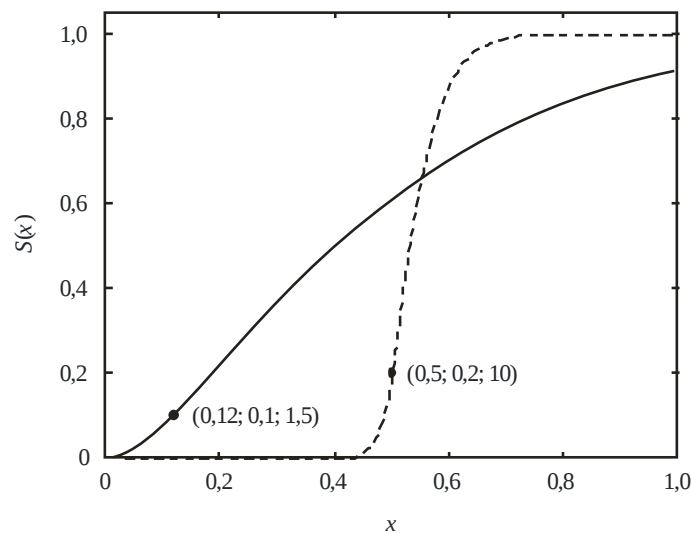
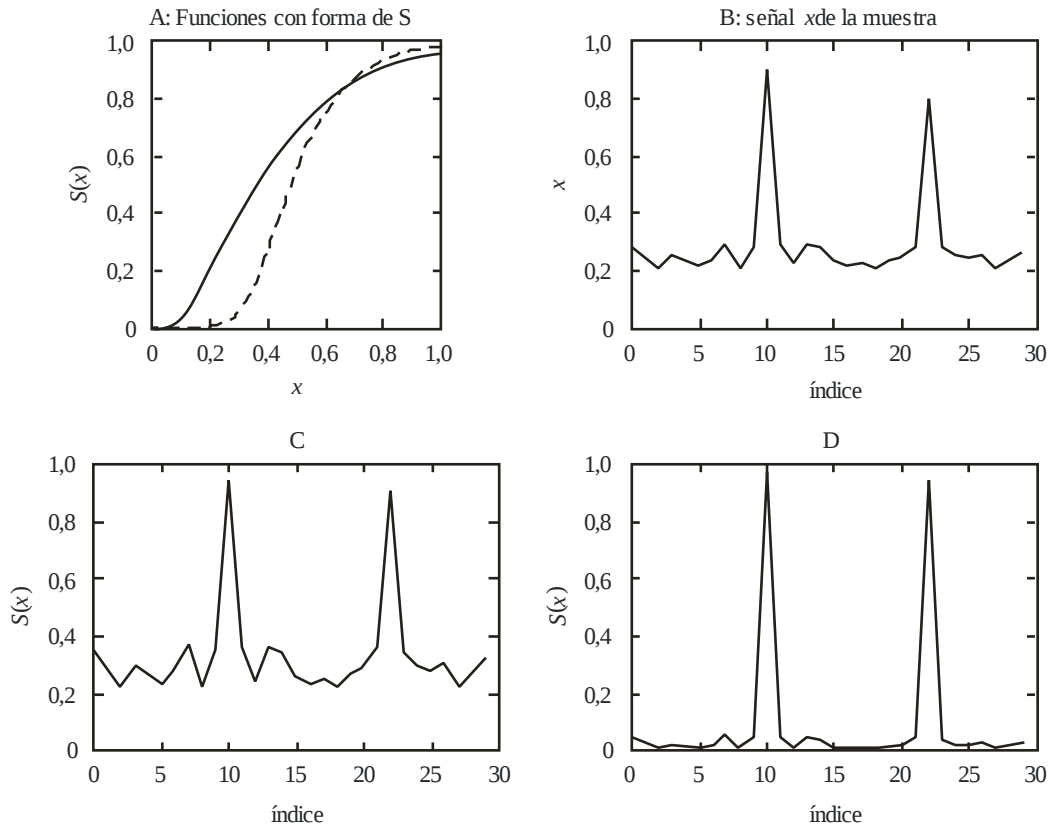


FIGURA 8

El gráfico (A) muestra las funciones con forma de S para dos conjuntos de parámetros distintos. El gráfico (B) muestra un vector de la señal de muestra. Todos los valores de x se transforman utilizando las dos funciones con forma de S de (A). El resultado se muestra en (C) y (D). La transformación con forma de S se utiliza para trasladar una característica x a una escala perceptual, a partir de parámetros que encajan con las muestras de datos, tal y como se indica en (C).

La utilización de una función con forma de S (línea discontinua en (A)) con parámetros dependientes de los datos el valor del primer parámetro es un múltiplo de la media de la señal de x , permite construir un detector de degradaciones transitorias, (D)



BT.1907-08

Utilizando una función con forma de S con parámetros constantes es posible trasladar el valor de la característica a una escala perceptual. Utilizando parámetros dependientes de los datos, la función con forma de S puede servir, por ejemplo, para comprimir todos los valores por debajo de la media en un valor pequeño o ampliar todos los valores superiores. De este modo, se puede emplear para calcular una característica sensible a degradaciones transitorias, véase la Fig. 8.

Siguiendo con la descripción del modelo, se define en primer lugar una degradación basada en las características de similitud *supra*, aumentando sin embargo el peso de las degradaciones más fuertes:

$$d_s = 1 - s_m + 1.5 s_{\text{delta}}.$$

A continuación, se utilizan dos funciones con forma de S, una de ellas con un conjunto fijo de parámetros cod_par , para transformar d_s en d_{cod} de acuerdo con una escala perceptual relacionada con una degradación básica, reflejando errores de codificación de vídeo.

Una segunda función con forma de S y parámetros *dependientes de los datos* permite transformar d_s en d_{trans} de acuerdo con una escala perceptual relacionada con degradaciones transitorias, reflejando de este modo errores de transmisión.

Más concretamente, en relación con el pseudocódigo *infra*, se emplea la función `SplitCodTrans`, utilizando como entrada el vector `d_s` y el vector de tiempo de exposición de trama `disp_time` y `d_cod`, `d_trans` como salida,

```
SplitCodTrans(d_s, disp_time, d_cod, d_trans).
```

El siguiente pseudocódigo muestra los detalles de la función. Cabe señalar que

```
stat.STransform(x,px,py,q)
```

denota la función con forma de S y tiene como entrada el valor real x que se transformará, y los tres parámetros denotados mediante (p_x, p_y, q) en la ecuación (8.1).

```
SplitCodTrans( const CVec& v, const CVec& dispTime,
               CVec& v_cod, CVec& v_trans ){

    // these parameters are determined empirically
    float qPosSmall = 0.55f;
    float qPosLarge = 0.65f;

    // q is the mean of the values in v between the qPosSmall and qPosLarge
    // quantiles.
    float q = r.TrimmedMean( qPosSmall, qPosLarge, v, dispTime );

    for( UINT i=0; i<v.Length(); i++ ){

        // the parameters used here are the result of fitting to sample
        // data
        v_cod(i) = stat.STransform(v(i), 0.07f, 0.1f, 2.0f);

        // Note that the STransform is not directly applied to v, but
        // to v(i)-q . This is preferred here for numerical reasons of the
        // resulting STransform.

        // v_trans is part of v above quantile value q
        v_trans(i) = vq_Max(0.0f, v(i)-q);

        float px = 0.5f * (q+0.2f);
        // the parameters used here are the result of fitting to sample
        // data
        v_trans(i) = stat.STransform(v_trans(i),px, 0.1f,16.0f);
    }
}
```

Véase `CQualityModelFullRef::SplitCodTrans` para todos los detalles relativos a la aplicación.

Análogamente, `d_diff_cod`, `d_diff_trans` se obtiene de `d_m`, `d_delta`, fijando

```
d_diff = d_m + 1.5 d_delta,
```

y calculando

```
SplitCodTrans(d_diff, disp_time, d_diff_cod, d_diff_trans),
```

a partir de los tres parámetros para las dos transformaciones con forma de S

```
cod_par = (4.0f,0.05f,0.2f)
trans_par = (0.5*(q+4.0f),0.1f,0.4f),
```

donde q es la media intercuantil, como en el pseudocódigo *supra*. Los resultados de la función son d_diff_cod , d_diff_trans .

A continuación, se calcula un valor característico relacionado con unos valores elevados de sacudida transitoria mediante una transformación con forma de S de la sacudida:

$$d_t_trans = S(\max(0, \text{jerkiness} - q))$$

siendo los parámetros de la transformación con forma de S los que vienen dados por $(\max(0.048, q), 0.2, 40.0)$ y q la media intercuantil entre los cuantiles 0,55 y 0,65 del vector de sacudida. Los parámetros se fijaron a partir de un gran conjunto de datos de muestra.

Véase `CQualityModelFullRef:: SplitTempTrans` para todos los detalles relativos a la aplicación.

A continuación se define la calidad básica q_cod como:

$$q_cod = (1 - d_cod) * (1 - d_diff_cod) * (1 - \text{blockiness}) \quad (8.2)$$

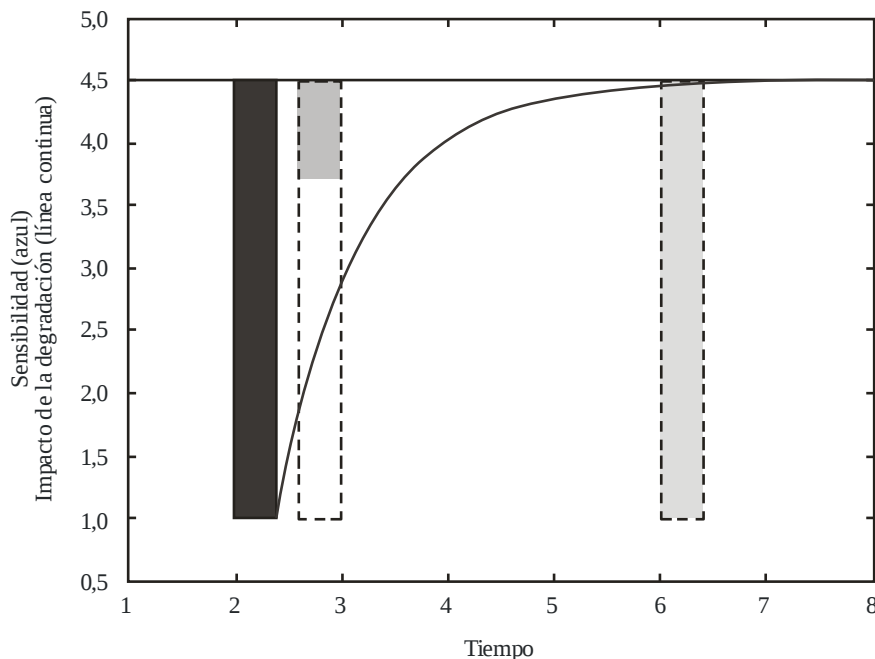
y la calidad transitoria como

$$q_trans = (1 - d_trans) * (1 - d_diff_trans) * (1 - d_t_trans).$$

La influencia de una degradación adicional se reduce si ocurre poco después de la primera degradación. Para tener en cuenta este efecto, se transforma q_trans en q_fq . La Fig. 9 ilustra esta idea.

FIGURA 9

La barra negra continua indica una degradación que se da en una secuencia de vídeo hipotética.
La línea negra continua indica la sensibilidad a degradaciones posteriores. La influencia de una degradación adicional, cuya fuerza viene dada por la línea discontinua, se reduce (primera barra de color gris oscuro) si ocurre poco después de la primera degradación.
La influencia de una degradación posterior no se ve alterada si se produce después de un largo intervalo temporal (barra de color gris claro de la derecha)



Los detalles del cálculo se explican mejor en el pseudocódigo de la siguiente sección, que utiliza $1-q_{trans}$ como vector de entrada v y $disp_time$ como vector de tiempo de exhibición de la trama. La constante de tiempo de caída $dT=1000$ ms se determinó a partir de datos de la muestra y posteriormente se fijó en

$$q_{fq} = 1 - w, \quad (8.3)$$

siendo w el vector de salida de la función:

```

DegFreq( const CVec& v, const CVec& disp_time, CVec& w, float dT ){

    // time constant for temporal integration of degradations
    float t_const = 80.0f;
    w(0) = v(0)*vq_Min(t_const,disp_time(0))/t_const;

    for( UINT i=1; i<v.Length(); i++ ){
        // integrate degradation over the last t_const milliseconds:
        // use an indicator function, which is 1 over the interval
        // [t_i-t_const, t_i] and 0 otherwise
        float dT_sum = 0.0f;
        UINT j=0;
        float v_sum = 0.0f;
        while( dT_sum < t_const && (int)i-(int)j>=0 ){
            // b is the overlap in [0,1] of the display time interval
            // of frame i-j with respect to the integration interval
            //      t_i-t_const   t_i
            //      |-----|
            //      |-----| integration window
            // -----> time
            // |   |   |   |   |   | frames
            //
            // ->| b*dT |<-
            //
            float b = vq_Min(t_const-dT_sum,disp_time(i-j)) / t_const;

            v_sum += v(i-j)*b;
            dT_sum += disp_time(i-j);
            j++;
        }
        // compute the fall-off factor a:
        float a = exp(-disp_time(i-1)/dT);
        // the degradation is now:
        // 1) a linear combination of the fall-off of a previous degradation
        // and the current (integrated) degradation, or
        // 2) the integrated current degradation (if it is stronger than 1)
    ).

    w(i) = vq_Max(v_sum, a * w(i-1) + (1.0f-a) * v_sum);

}

```

Proceso de promediación

$\text{disp_time}(i)$ denota el tiempo de exhibición de la trama i y $\text{jerkiness}(i)$, el valor de sacudida correspondiente a la trama i . Se recupera q_{cod} de la ecuación (8.2) y q_{fq} de la ecuación (8.3). Se fija

$$\begin{aligned} Q_{\text{t}} &= 1 - 1/T \sum_i \text{jerkiness}(i) \\ Q_{\text{fq}} &= 1/T \sum_i q_{\text{fq}}(i) * \text{disp_time}(i) \\ Q_{\text{cod}} &= 1/T \sum_i q_{\text{cod}}(i) * \text{disp_time}(i) \end{aligned}$$

siendo $T = \sum_i \text{disp_time}(i)$, y $\sum_i$ la suma en todos los índices $i=0, \dots, \text{number_of_frames}$.

El valor final previsto es un producto y se ajusta a la gama de MOS [1,5]:

$$s = 4 * Q_{\text{t}} * Q_{\text{cod}} * Q_{\text{fq}} + 1$$

Véase el método `CQualityModelFullRef::PredictScoreCodTrans` para los detalles relativos a la predicción del resultado.

2.9 Manejo de secuencias de vídeo con desalineamientos espaciales importantes

Para evitar errores de predicción en casos de desalineamientos espaciales relativamente importantes entre las secuencias de vídeo procesada y de referencia, se calculan los pasos anteriormente mencionados para tres pasos distintos de la alineación espacial horizontal y vertical de la secuencia de vídeo; se considerará que el resultado previsto máximo de todas las posiciones espaciales es el valor final estimado de la calidad.

En cada dirección se utiliza un tamaño de paso de cuatro píxeles. De este modo, se obtiene una gama de búsqueda espacial de ± 8 píxeles, que abarca fácilmente el desplazamiento espacial máximo del conjunto de la prueba (± 5 píxeles). Dado que esta ampliación se lleva a cabo en una función de alto nivel del modelo, la gama de alineación puede adaptarse fácilmente a desplazamientos más grandes o puede reducirse (por ejemplo, ± 4 píxeles) para ahorrar recursos de cálculo.

Véase `vquad_hd::vq_vquad08`.

2.10 Aplicación del código fuente de referencia

Esta sección contiene un archivo integrado con el código fuente en lenguaje C++ que abarca las partes y las funciones esenciales para una aplicación conforme a la descripción del modelo. Todos los enlaces a las aplicaciones reales que figuran en las secciones anteriores se refieren a este código fuente de referencia.

Anexo 3

Prueba de conformidad

Este Anexo se completa con un archivo digital adjunto que contiene la información y los archivos siguientes:

- 1) Dieciséis secuencias breves de vídeo en alta definición (secuencias de referencia y degradadas para ocho casos prácticos). Estas secuencias abarcan distintas distorsiones y contenidos. Se ofrecen para que el usuario de la presente Recomendación pueda realizar una prueba de conformidad del modelo con respecto a su aplicación de referencia. Las secuencias de vídeo se componen únicamente de algunas tramas, para reducir la capacidad de almacenamiento. La finalidad de dichas secuencias no es llevar a cabo pruebas visuales, sino simplemente una prueba de conformidad de la aplicación del modelo de alta definición.
- 2) Valores previstos de MOS para secuencias de alta definición mencionadas en 1) como 'HD_ConformanceReferenceResults.xls'. Estos valores se obtuvieron de una aplicación de referencia del modelo de alta definición.
- 3) Valores previstos de MOS para cinco bases de datos públicas en alta definición disponibles a través del VQEG. Estas bases de datos pueden emplearse como prueba de conformidad ampliada para las aplicaciones del modelo.

Criterios de la prueba de conformidad:

- i) Los ocho valores de referencia que se dan en 2) deben reproducirse exactamente mediante una aplicación del modelo. «Exactamente» significa que debe reproducirse el valor con una resolución de tres dígitos decimales.
- ii) Los valores previstos de MOS de las cinco bases de datos públicas del VQEG deben reproducirse con una desviación muy limitada. Están permitidas variaciones menores, ya que la experiencia ha demostrado que distintas optimizaciones en materia de velocidad y uso del almacenamiento pueden conducir a desviaciones menores y negligibles en el resultado final.

CUADRO 2

Distribución permitida de las diferencias en los datos de la prueba de conformidad

Diferencia absoluta	Ocurrencia permitida
> 0,0001	5,00%
> 0,001	1,00%
> 0,01	0,50%
> 0,1	0,05%
> 0,3	0,00%

En bases de datos distintas de las definidas en el presente Anexo 3, no debe rebasarse la misma distribución del error. Para datos desconocidos, debe tomarse para estas estadísticas un conjunto de prueba de al menos 500 pares de archivos, preferiblemente procedentes de experimentos totalmente subjetivos.

Bibliografía

- Informe final sobre la prueba de validación de la Fase I de TVAD del VQEG (2010), «*Video Quality Experts Group: report on the validation of video quality models for high definition video content*», Grupo de Expertos en calidad del vídeo (VQEG), <http://www.its.bldrdoc.gov/vqeg/projects/hdtv>.
- Recomendación UIT-R BT.500-12 (2009), Metodología para la evaluación subjetiva de la calidad de las imágenes de televisión.
- Recomendación UIT-R BT.601-7 (2011), Parámetros de codificación de televisión digital para estudios con formatos de imagen normal 4:3 y de pantalla ancha 16:9.
- Recomendación UIT-T H.264 (2012), Codificación de vídeo avanzada para los servicios audiovisuales genéricos.
- Recomendación UIT-T J.143 (2000), Requisitos de usuario para mediciones objetivas de la percepción de la calidad vídeo en televisión digital por cable.
- Recomendación UIT-T J.144 (2004), Técnicas de medición objetiva de la percepción de la calidad vídeo en televisión por cable en presencia de una referencia completa.
- Recomendación UIT-T J.148 (2003), Requisitos para un modelo objetivo de calidad perceptual de los multimedios.
- Recomendación UIT-T J.149 (2004), Método para especificar la precisión y la calibración cruzada de métricas de la calidad de vídeo.
- Recomendación UIT-T J.244 (2008), Métodos para calibrar la desalineación constante de los dominios espacial y temporal con ganancia y desplazamiento constantes.
- Recomendación UIT-T J.247 (2008), Medición objetiva de la calidad de vídeo multimedios percibida en presencia de una referencia íntegra.
- Recomendación UIT-T J.340 (2010), Algoritmo de referencia para calcular la relación señal/ruido de cresta (PSNR) de una secuencia vídeo procesada con desplazamientos espaciales constantes y un retardo constante.
- Recomendación UIT-T P.910 (2008), Métodos de evaluación subjetiva de la calidad vídeo para aplicaciones multimedios.
- Recomendación UIT-T P.911 (1998), Métodos de evaluación subjetiva de la calidad audiovisual para aplicaciones multimedios.
- Recomendación UIT-T P.931 (1998), Retardo de las comunicaciones multimedios, sincronización y medición de la velocidad de tramas.
-