UNIVERSIDAD MIGUEL HERNÁNDEZ DE ELCHE

TESIS DOCTORAL

Técnicas de codificación perceptual para compresores de vídeo de última generación

Javier Ruiz Atencia

2024

Director: Miguel Onofre Martínez Rach

Programa de Doctorado en

Tecnologías Industriales y de Telecomunicación

RELACIÓN DE TRABAJOS PUBLICADOS

La presente tesis doctoral, titulada "Técnicas de codificación perceptual para compresores de vídeo de última generación", se presenta bajo la modalidad de **tesis por compendio** de las siguientes **publicaciones**:

- **J. R. Atencia**, O. L. Granado, M. P. Malumbres, M. O. Martínez-Rach and G. Van Wallendael, "Analysis of the Perceptual Quality Performance of Different HEVC Coding Tools," in IEEE Access, vol. 9, pp. 37510-37522, 2021, doi: 10.1109/ACCESS.2021.3062938.
  - **(Q1)** Scimago (Scopus- Scimago Journal & Country Rank).
  - Incluido totalmente en la tesis (capítulo 3).

- **J. R. Atencia**, O. L. Granado, M. P. Malumbres, M. O. Martínez-Rach, D. Ruiz-Coll, G. Fernández-Escribano and G. Van Wallendael, "A Hybrid Contrast and Texture Masking Model to Boost HEVC Perceptual RD Performance," in Electronics, vol. 13, n. 16, 2024, doi: 10.3390/electronics13163341.
  - **(Q2)** JCR (Clarivate Journal Citation Reports).
  - Incluido totalmente en la tesis (capítulo 4).

INFORME DEL DIRECTOR DE LA TESIS

El Dr. D. Miguel Onofre Martínez Rach, profesor titular de la Universidad Miguel Hernández de Elche,

CERTIFICA

que la presente tesis titulada "Técnicas de codificación perceptual para compresores de vídeo de última generación" ha sido realizada bajo su dirección, en el Departamento de Ingeniería de Computadores de la Universidad Miguel Hernández de Elche por D. Javier Ruiz Atencia y autoriza su presentación por la modalidad de compendio de publicaciones ante la Comisión de Doctorado de la Universidad Miguel Hernández de Elche.

Y para que así conste, a todos los efectos oportunos, firmamos el presente informe.

Director

Dr. Miguel Onofre Martínez Rach

INFORME DEL COORDINADOR DE LA COMISIÓN ACADÉMICA DEL PROGRAMA DE DOCTORADO

El Dr. D. Óscar Reinoso García, Coordinador del Programa de Doctorado en Tecnologías Industriales y de Telecomunicación,

INFORMA

Que D. Javier Ruiz Atencia ha realizado bajo la supervisión de nuestro Programa de Doctorado el trabajo titulado «Técnicas de codificación perceptual para compresores de vídeo de última generación», conforme a los términos y condiciones definidos en su Plan de Investigación y de acuerdo al Código de Buenas Prácticas de la Universidad Miguel Hernández de Elche, cumpliendo los objetivos previstos de forma satisfactoria para su defensa pública como tesis doctoral.

Y para que así conste, a todos los efectos oportunos, firmo el presente informe.

Coordinador

Dr. Óscar Reinoso García

AGRADECIMIENTOS

La culminación de esta tesis doctoral ha sido un largo y desafiante recorrido que no habría sido posible sin el apoyo incondicional y la paciencia de muchas personas a lo largo de estos años.

Quisiera expresar mi más sincero agradecimiento a mi director de tesis, Miguel Onofre Martínez-Rach y a mi tutor Manuel Pérez Malumbres, por su guía, tutorización, y sobre todo, por la paciencia que han tenido conmigo en cada paso del proceso. Su orientación constante, sus valiosas sugerencias, y su disposición a resolver cada duda han sido fundamentales para llegar hasta aquí.

También me gustaría reconocer al resto del departamento de Arquitectura de Computadores, en especial a Otoniel López Granado, quien ha compartido conmigo su experiencia y conocimientos, además de soportar algún que otro quebradero de cabeza que le he dado. Su apoyo ha sido crucial, sobre todo en los momentos más complicados.

A mi pareja, Antonia, gracias por tu infinita paciencia y por cuidar de Unai y de mí a lo largo de esta aventura. Sin tu apoyo, comprensión y amor incondicional, este camino habría sido mucho más arduo. A nuestro pequeño Unai, que llegó en medio de esta travesía, gracias por recordarme cada día lo que realmente importa en la vida.

Por último, a mis padres, quienes desde el principio me animaron a estudiar y seguir adelante en cada etapa académica. Sin vuestro aliento constante, no habría llegado hasta aquí. Gracias por creer en mí desde siempre.

RESUMEN

Esta tesis aborda el desarrollo de técnicas avanzadas de codificación perceptual aplicadas al estándar de codificación de vídeo HEVC (H.265). En un contexto de creciente demanda de contenido audiovisual en alta resolución, resulta crucial desarrollar algoritmos de compresión más eficientes que mantengan la calidad visual percibida por el espectador.

En el primer bloque se investiga en profundidad el rendimiento de diversas herramientas de codificación presentes en el estándar HEVC desde una perspectiva perceptual. Utilizando métricas objetivas como la SSIM, MS-SSIM y PSNR-HVS-M, se comparan diferentes técnicas de codificación, evaluando tanto su impacto en la calidad visual subjetiva como su eficiencia en la compresión. El trabajo revela importantes hallazgos sobre cómo optimizar la relación tasa-distorsión (Rate-Distortion, R/D), sentando las bases para mejoras futuras en la codificación perceptual.

El segundo bloque de la tesis introduce una metodología que combina modelos de sensibilidad al contraste y enmascaramiento por textura para ajustar dinámicamente los parámetros de cuantificación (QP) en función del contenido visual. Este enfoque híbrido permite una mayor adaptación del codificador a las características de la escena, maximizando la eficiencia de compresión en las regiones de mayor relevancia perceptual, preservando los detalles visuales más importantes para el espectador.

Finalmente, en las conclusiones, se resumen las principales contribuciones de la tesis, destacando los avances en la codificación perceptual y sus implicaciones en el futuro de los estándares de vídeo. Asimismo, se proponen líneas de desarrollo posteriores, orientadas hacia la implementación en tiempo real y la generalización de los modelos presentados a otros estándares de compresión de vídeo.

En conjunto, las contribuciones de esta tesis representan avances significativos en la codificación de vídeo perceptual, destacando mejoras en la eficiencia de compresión sin comprometer la calidad visual. Asimismo, se proponen líneas de desarrollo futuras, orientadas hacia la implementación en tiempo real, el uso de redes neuronales y la generalización de los modelos presentados a otros estándares de compresión de vídeo más recientes.

ABSTRACT

This thesis addresses the development of advanced perceptual coding techniques applied to the HEVC (H.265) video coding standard. In the context of increasing demand for high-resolution audiovisual content, it is crucial to develop more efficient compression algorithms that maintain the visual quality perceived by the viewer.

In the first section, the performance of various coding tools present in the HEVC standard is investigated in depth from a perceptual perspective. Using objective metrics such as SSIM, MS-SSIM, and PSNR-HVS-M, different coding techniques are compared, evaluating both their impact on subjective visual quality and their efficiency in compression. The study reveals important findings on how to optimize the rate-distortion (R/D), laying the groundwork for future improvements in perceptual coding.

The second section of the thesis introduces a methodology that combines contrast sensitivity models and texture masking to dynamically adjust the quantization parameters (QP) based on visual content. This hybrid approach allows the encoder to better adapt to scene characteristics, maximizing compression efficiency in regions of higher perceptual relevance, preserving the most important visual details for the viewer.

Finally, in the conclusions, the main contributions of the thesis are summarized, highlighting the advances in perceptual coding and their implications for the future of video standards. Additionally, future development lines are proposed, focusing on real-time implementation and the generalization of the presented models to other video compression standards.

Overall, the contributions of this thesis represent significant advances in perceptual video coding, highlighting improvements in compression efficiency without compromising visual quality. Furthermore, future development paths are proposed, focusing on real-time implementation, the use of neural networks, and the generalization of the models presented to more recent video compression standards.

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

1. PRÓLOGO

1.1. Motivación

La industria audiovisual está experimentando un notable aumento en la demanda y distribución de contenidos, especialmente en el formato de vídeo. De acuerdo con los informes más recientes [1], en 2022, el 65 % del tráfico de Internet corresponde a vídeo en diversas formas, lo que significa más de un millón de minutos de vídeo por segundo que viajan simultáneamente por la red. Este incremento se atribuye, en parte, al aumento en el tamaño y la resolución de los paneles de televisión en los últimos años. Se estima que un 66 % de las pantallas de TV vendidas en 2023 tendrán una resolución 4K [2], lo que supone un aumento significativo respecto al año 2018. Adicionalmente, existe una tendencia creciente hacia el consumo de contenidos multimedia en diversos dispositivos y en cualquier lugar, subrayando la necesidad de adaptabilidad y portabilidad.

Esta tendencia está transformando todos los sectores de la industria audiovisual, generando nuevos modelos de negocio y convergiendo las plataformas de distribución tradicionales hacia tecnologías basadas en IP. Los fabricantes de dispositivos están adoptando estándares para acelerar la implementación de nuevas tecnologías y mejorar la experiencia del usuario.

El presente proyecto de tesis se alinea precisamente con esta evolución de la industria audiovisual. Se propone desarrollar técnicas que faciliten una distribución eficiente, robusta y con una alta Quality of Experience (QoE), en términos de calidad perceptual, para los contenidos en plataformas basadas en IP. Se proponen innovadoras técnicas y prototipos de software que abordan los desafíos presentes y futuros de la industria audiovisual. Estos prototipos permitirán evaluar y mejorar la calidad perceptual del vídeo recibido en redes de transporte basadas en IP, maximizando la experiencia del usuario al visualizar secuencias de vídeo con distintas resoluciones, desde definición estándar (SD) hasta Ultra alta definición (UHD), utilizando técnicas de codificación perceptual para los últimos estándares de vídeo.

Los resultados de esta investigación representan innovaciones tecnológicas con un al-

to grado de transferencia a la industria, tanto en los sistemas integrados en la producción y distribución de contenidos (YouTube, Netflix, Disney+, Prime, etc.), como en los dispositivos comerciales dirigidos a la distribución de contenidos en plataformas IP (codificadores/transcodificadores) y en los dispositivos finales de usuario (teléfonos, televisores, relojes inteligentes, tabletas, PCs, sistemas de realidad virtual (VR) y realidad aumentada (AR), Display Walls, etc.), mejorando así la calidad perceptual del vídeo entregado y ofreciendo una experiencia de visualización óptima, en cualquier dispositivo y en cualquier lugar.

1.2. Objetivos

Este proyecto de esta tesis tiene como objetivo global desarrollar técnicas que permitan mejorar la calidad perceptual de los contenidos de vídeo en HD (High Definition) / UHD (Ultra High Definition), para su distribución eficiente y robusta en plataformas basadas en IP utilizando los últimos estándares de vídeo. Para alcanzar este objetivo global, se han definido los siguientes objetivos específicos:

- **Desarrollo de software de prototipado.** Se propone diseñar y programar en Matlab las principales etapas del estándar de codificación de vídeo HEVC, en modo Intra, para servir como software de prototipado. Este software permitirá analizar y validar eficientemente distintas alternativas de codificación perceptual de manera eficiente. Contará con dos modos de funcionamiento, HEVC y PHEVC (Perceptual HEVC). Las secuencias generadas en el modo HEVC serán completamente idénticas a las obtenidas por el software de referencia, mientras que en el modo PHEVC se permitirá la inclusión de diferentes técnicas perceptuales. El software contará con un sistema de generación de curvas Rate-Distortion (R/D), teniendo como valores de distorsión métricas de calidad objetivas como la SSIM, MS-SSIM, VIF, PSNR-HVS-M, entre otras [3]-[6]. Además, el software exportará de manera automática métricas de rendimiento utilizando el método Bjøntegaard [7], que permite calcular el ahorro en bit-rate (BD-Rate) o las mejoras en calidad (BD-SSIM, BD-MS-SSIM, etc.) que se obtienen para cada codificación.

- **Evaluación perceptual de herramientas de codificación.** Se proyecta realizar un

análisis perceptual exhaustivo de las herramientas de codificación (coding tools) incluidas en el software de referencia del estándar de vídeo HEVC. Aunque tradicionalmente la eficiencia de estas herramientas ha sido evaluada mediante métricas puramente objetivas, como la MSE o la PSNR, se reconoce la limitación de estas medidas al no considerar la subjetividad inherente al sistema visual humano [3], [4], [8], [9]. Este objetivo se centrará en evaluar la respuesta perceptual de dichas herramientas de codificación mediante métricas de calidad objetivas como la SSIM, MS-SSIM y PSNR-HVS-M, para determinar cuáles proporcionan una mejora en la calidad, en términos perceptuales.

- **Estudio e integración de técnicas de codificación perceptual.** Se pretende explorar y evaluar diferentes técnicas de codificación en el estándar HEVC, como las basadas en la curva de sensibilidad al contraste (CSF), que permite discriminar las frecuencias espaciales para las que el ojo humano es más sensible y, por tanto, descartar información no relevante en otras frecuencias. También se evaluarán técnicas de enmascaramiento por luminancia y por textura. El enmascaramiento es una propiedad del sistema visual que consiste en que un estímulo local, por ejemplo, la luminancia en una zona concreta de la imagen oculta distorsiones o impide la detección de frecuencias espaciales específicas en la misma zona. Analizando la luminancia y textura en las distintas zonas de la imagen y midiendo la cantidad de enmascaramiento producido, se puede determinar cuánta información visual es susceptible de ser eliminada sin ser detectada por el sistema visual, lo cual podría reducir el bit-rate manteniendo la misma calidad perceptual. Se investigarán técnicas de cuantificación perceptual que permitan caracterizar y parametrizar el cuantificador en función del grado de detección perceptual de cada una de las técnicas estudiadas.

- **Desarrollo de un clasificador de bloques.**: Se planea desarrollar un clasificador de bloques en los que se divide una imagen durante su codificación basado en su nivel de textura, diferenciando entre bloques suaves o planos, bloques con bordes y bloques con un alto nivel de textura o variabilidad en sus píxeles. Se creará una base de datos de bloques atendiendo a los diferentes tamaños en los que se puede particionar un bloque en el estándar HEVC, y se diseñará una interfaz gráfica en Matlab para facilitar el proceso de clasificación manual de los bloques, generando

un fichero CSV con el fin de ser utilizado como entrada a algoritmos de clasificación supervisada para construir un modelo clasificador para usar en las técnicas perceptuales propuestas.

- **Integración en el Software de referencia.** Se tiene como meta implementar y validar las técnicas perceptuales más eficaces en el software de referencia HM, programado en C++. Esto permitirá una integración normalizada y accesible para la comunidad investigadora, y se evaluará la eficiencia computacional comparada con el prototipo en MATLAB.

1.3. Publicaciones

Las contribuciones publicadas en las que se apoya esta tesis corresponden a dos artículos en revistas indexadas, una de ellas en el primer cuartil (Q1) del índice Scimago (Scopus - Scimago Journal & Country Rank) y la otra en el segundo cuartil (Q2) del Journal Citation Reports (JCR). Estas publicaciones, cuyos metadatos se proporcionan a continuación, están directamente alineadas con el propósito y objetivo de esta tesis.

- Analysis of the Perceptual Quality Performance of Different HEVC Coding Tools [10].
  J. R. Atencia, O. L. Granado, M. P. Malumbres, M. O. Martínez-Rach and G. Van Wallendael
  IEEE Access. Vol. 9, pp. 37510-37522 (2021)
  SJR Impact Factor: 0.927, Quartile **Q1**.
  DOI: 10.1109/ACCESS.2021.3062938.

- A Hybrid Contrast and Texture Masking Model to Boost HEVC Perceptual RD Performance [95].
  J. R. Atencia, O. L. Granado, M. P. Malumbres, M. O. Martínez-Rach, D. Ruiz-Coll, G. Fernández-Escribano and G. Van Wallendael
  Electronics. Vol. 13, n. 16 (2024),
  JCR Impact Factor: 0.644, Quartile **Q2**.
  DOI: 10.3390/electronics13163341

2. INTRODUCCIÓN

En los últimos años ha ido tomando cada vez más importancia la calidad que el usuario percibe en la reproducción de los contenidos audiovisuales. Para medir la calidad visual de un vídeo existen diversas métricas de calidad objetiva (PSNR, MSE) que han venido utilizándose habitualmente en la literatura. Sin embargo, es conocido que, bajo ciertas circunstancias, la valoración de la calidad visual que percibe un usuario no se correlaciona con estas métricas objetivas, pudiendo llegar a ofrecer valoraciones contrapuestas [15]. En este sentido, en los procesos de codificación se están integrando técnicas, mecanismos y métricas que tratan de maximizar la calidad perceptual que aprecia el usuario cuando se decodifica y reproduce el contenido recibido [16]. Entre otras, (1) las técnicas basadas en la sensibilidad al contraste (Contrast Sensitivity Function – CSF) del sistema visual humano, (2) técnicas de enmascaramiento por luminancia, textura, saliency (atención), enfoque, etc., que permiten determinar la información redundante (local, espacial/temporal) para nuestro sistema visual, y (3) la cuantificación perceptual, capaz de descartar primero la información que perceptualmente es menos importante para el usuario.

Las técnicas que aplican la sensibilidad al contraste han sido ampliamente utilizadas en la literatura. Respecto a su aplicación en los últimos estándares, existe una clasificación [17] en dos categorías, técnicas con una aproximación basada en modelos de visión como son el procesamiento de vídeo basado en regiones de interés y modelos de atención [18], [19] y técnicas con una aproximación basada en el procesamiento de la señal como procesamiento guiado por métricas de calidad [20]-[22], codificaciones basadas en la sensibilidad al contraste del sistema visual [23]-[25], en la textura de la escena [26]-[28] y en la optimización de redundancia espacial [29].

En relación con las técnicas de enmascaramiento por luminancia y textura, los modelos basados en Just-Noticeable-Difference (JND) definen la menor diferencia detectable entre dos señales. Como el propósito de la codificación de vídeo es obtener la mejor calidad perceptual, los umbrales JND se usan para determinar los tamaños óptimos de los pasos de cuantificación. Algunos trabajos, como [30]-[32] utilizan la técnica JND aplicada al estándar de vídeo HEVC. Kim *et al.* [32] usan JND para el modo TSM (Transform

Skip Mode) y para el modo no-TSM ajustando los parámetros del RDO (Rate Distortion Optimization). Para el modo TSM utilizan el modelo JND basado en luminance-masking mientras que para el modo no-TSM utilizan JND basado en temporal-masking, contrast-masking, y usan la CSF para la luminancia. Yang *et al.* [30] proponen un método de RDO basado en JND y SAD (Sum of Absolute Difference). Ndjiki-Nya *et al.* [31] han usado una codificación de vídeo basada en el contenido utilizando análisis de texturas y en [33] los autores han aplicado estas técnicas al HEVC. Usando esta técnica el codificador puede saltar regiones cuya textura ya ha sido analizada y almacenada, y el decodificador puede reconstruir dichas regiones accediendo a ese almacén de texturas. Adzic *et al.* [34] proponen una codificación perceptual temporal. Basándose en la relación entre la agudeza visual al contraste en función de la velocidad retinal, se puede desarrollar un modelo temporal de agudeza basándose en los vectores de movimiento de un bloque y su contenido frecuencial.

Los modelos de atención del espectador también son usados para determinar información irrelevante, tanto espacial como temporal, de forma que permiten decidir qué información y en qué grado poder prescindir de la misma para conseguir ahorrar bit-rate manteniendo una buena calidad de experiencia o calidad visual por parte del espectador. Es bien sabido que cuando las personas miran un vídeo o una imagen solo pueden prestar atención a una región u objeto en particular de la escena y no a la imagen completa. La atención del observador en una imagen viene definida por muchos factores, clasificables como visuales y semánticos. Los factores semánticos vienen definidos por la experiencia, compresión e interpretación de la escena (imagen estática) que tiene el observador. Un ejemplo de ello es la atención prestada a las caras, independientemente de su perfil, desenfocadas o incluso parcial, o completamente tapadas, pues el sistema visual sabe que ahí hay una cara. Las técnicas basadas en ROI han sido utilizadas en otros codificadores de vídeo, sin embargo, como H.265/HEVC presenta una nueva estructura de codificación basada en árboles cuaternarios (Quad-tree), la aplicación de ROI debe ser adaptada a este nuevo esquema. En [35], [36], se propone un esquema de asignación de bits basado en un modelo de percepción jerárquica de la cara. Considerando que los ojos, la boca y otras áreas de la cara son de diferentes niveles de interés, se establecen diferentes pesos para las diferentes regiones de la cara. Itti *et al.* [37] han propuesto un modelo de atención visual neurológico para seleccionar regiones de alto impacto visual (“prominencia”). En

este modelo se proponen 63 variaciones de este algoritmo con el fin de obtener un balance entre la calidad obtenida y el bit-rate necesario. Tomando estos resultados de investigación en el marco H.265/HEVC, Li *et al.* [38] utilizan el mapa de prominencia calculado a partir del modelo de atención visual computacional para ajustar la asignación de bits para obtener una mejor calidad perceptual. Milani *et al.* [39] usan un algoritmo de detección de objetos para generar una métrica de prominencia y optimizar la asignación de bits en los bordes de los objetos. Como se muestra en estos dos métodos, la idea es aumentar o disminuir los pasos de cuantificación según las probabilidades de las unidades de codificación indicadas por el mapa de prominencia. De esta forma, podemos preservar los detalles de las regiones de atención (ROA) y reducir el bit-rate cuantificando más el resto de las regiones.

Entre los factores que dependen estrictamente de la información visual, sin componente semántica, uno de los más importantes es el enfoque de escena. Cuando una escena tiene una o varias áreas enfocadas respecto al resto (background), inconscientemente el observador sitúa su atención en las áreas enfocadas de la misma. Liang *et al.* [40] proponen una codificación de vídeo perceptual basada en la escena mediante la reconstrucción de escena para reconocer el área de primer plano y el área de fondo. Para proteger los bordes de los objetos los autores proponen utilizar la correspondiente información de distancia para ajustar el valor QP con el fin de obtener la mejor calidad subjetiva. En [41] los autores construyen un Quad-tree guiado por la calidad del enfoque de cada bloque, dividiendo el bloque si el mismo resultan simultáneamente áreas enfocada y no enfocadas. Esta técnica segmenta la región a proteger de la pérdida de información producida en la cuantificación. Uno de los problemas fundamentales es determinar el área enfocada mediante el uso de una métrica del enfoque apropiada. En [42] los autores realizan un extenso estudio de las distintas métricas de enfoque utilizadas en la literatura valorando aspectos como la calidad, el tiempo y la complejidad de estas. En [43] los autores realizan un análisis R/D basado en métricas de desenfoque con lo que determinan los valores QP para cada bloque. También abordan otros de los puntos fundamentales, que es medir la calidad resultante teniendo en cuenta la zona de interés definida por un eye-tracker y la calidad general de la imagen, obteniendo ganancias en bit-rate en ambos casos. Las distintas técnicas mencionadas utilizan distintos grados de cuantificación para proporcionar la correcta relación entre ahorro de bit-rate y calidad perceptual, donde la cuantificación

perceptual puede parametrizarse utilizando cuantificadores adaptativos o variables como los vistos en [21], [24].

En cuanto a la valoración del rendimiento desde un punto de vista perceptual, las métricas objetivas de valoración de la calidad permiten acercar la valoración subjetiva humana a un valor numérico de calidad calculado por dichas métricas. Por ejemplo, el índice de similitud estructural (SSIM) nos permite una mejor evaluación de la calidad perceptual que usando la relación señal-ruido (PSNR). En algunos trabajos, la métrica SSIM se utiliza en el proceso de optimización del rate-distortion (RDO). Yeo *et al.* [44] proponen un RDO basado en la SSIM para HEVC. Ellos calculan la relación entre SSIM y MSE, y utilizan el valor de SSIM para reemplazar el MSE en el proceso de rate-distortion. En otros trabajos, como en [45], utilizan un esquema similar basado en SSIM y una normalización por división de coeficientes de la transformada discreta de coseno (DCT), donde el factor de división de los coeficientes se obtiene midiendo la energía de los coeficientes vecinos.

2.1. Revisión histórica de los algoritmos de codificación de vídeo

La compresión digital de vídeo desempeña un papel fundamental en el almacenamiento y la transmisión de contenido multimedia. La cantidad de datos transmitidos a través de Internet se duplica cada año, y un elevado porcentaje de esos datos corresponden a vídeo [1]. Reducir las necesidades de ancho de banda de cualquier dispositivo resultará en reducciones significativas de costes y hará que el dispositivo sea más asequible. La compresión digital de vídeo ofrece formas de representar un vídeo de manera más compacta, permitiendo aumentar el almacenamiento y reduciendo los costes y tiempos de transmisión. Las ventajas de la compresión de vídeo conllevan un coste computacional asociado. Antes de almacenar o transmitir una secuencia de vídeo, se debe codificar para reducir el número de bits necesarios para su reproducción.

Históricamente, dos principales organismos han liderado el desarrollo de estándares de codificación de vídeo: la International Telecommunications Union (ITU-T) y la International Standardization Organization/International Electrotechnical Commission (ISO/IEC) a través del grupo Moving Picture Experts Group (MPEG). Mientras que ITU-T se centró inicialmente en aplicaciones de transmisión en tiempo real, MPEG estaba más orientado

hacia la distribución y emisión de vídeo.

Uno de los primeros estándares notables fue el ITU-T H.261 [46], lanzado en 1988, diseñado específicamente para videoconferencias y que introdujo la arquitectura de codificación híbrida. La utilización de macrobloques (MB) de 16×16 píxeles como unidad de codificación, el uso de predicción por compensación de movimiento (MC) a partir de vectores de movimiento (MV) o la codificación diferencial residual utilizando la transformada discreta del coseno (DCT) con algunas de las herramientas fundamentales introducidas en este estándar que fueron adoptadas por estándares posteriores.

Posteriormente, MPEG presentó el estándar MPEG-1 [47], enfocado a la distribución y emisión de vídeo. Este estándar, basado en el mismo esquema de codificación que H.261, incluye herramientas adicionales como la compensación de movimiento bidireccional o la búsqueda rápida hacia adelante y atrás. Se utilizó ampliamente para Vídeo CD y desempeñó un papel fundamental en la representación de datos multimedia.

En 1995, la colaboración entre ITU-T y MPEG dio lugar al estándar conjunto MPEG-2/H.262 [48], [49], que extendió las capacidades de MPEG-1 para soportar vídeo entrelazado y escalabilidad. Se utilizó principalmente para la transmisión (satélite y terrestre) de televisión digital y sistemas HDTV y el DVD, siendo usado como referencia para posteriores estándares de codificación.

El ITU-T H.263 [50] surgió en 1996 como una mejora del H.261, optimizando todavía más la transmisión de vídeo en tiempo real para tasas de bits relativamente bajas. Entre sus mejoras se incluyen una estimación de movimiento más eficiente, herramientas de codificación negociables y el concepto de perfiles y niveles.

Sin embargo, uno de los saltos más significativos en la eficiencia de compresión se logró con la colaboración en 2003 de ITU-T e ISO/IEC en el llamado Joint Video Team Video Coding (JVT-VC), cuando presentaron el H.264/AVC [51], [52]. Este estándar mantenía el mismo esquema de codificador híbrido, pero introdujo numerosas herramientas de codificación avanzadas, logrando reducciones significativas en las tasas de bits.

A medida que avanzaba la década, surgió la necesidad de lograr mayores niveles de compresión. Esta demanda condujo a la formación del Joint Collaborative Team on Video Coding (JCT-VC) en 2010, dedicada a investigar diferentes métodos con el fin de obtener una reducción de la tasa de bits del 50 % con respecto al estándar H.264/AVC, y que

finalmente lanzó H.265/High Efficiency Video Coding (HEVC) [53], [54] en 2013. Este estándar abordaba, entre otras, la demanda de mayor resolución de vídeo y la aplicación de técnicas de procesamiento en paralelo, al tiempo que mejoraba significativamente la eficiencia de compresión en comparación con su predecesor. El concepto de macrobloques es sustituido por los Coding Tree Units (CTUs) de hasta 64×64 píxeles, pudiendo ser particionado siguiendo una estructura de árbol jerárquico denominada QuadTree. Los modos de predicción intraframe son aumentados hasta 35, así como la inclusión de diferentes filtros de postprocesamiento. Todo esto lograba ahorros de aproximadamente el 59 % en tasa de bits con respecto a su predecesor, manteniendo una calidad subjetiva similar. Como veremos más adelante, este nuevo estándar cumple con los requisitos planteados por la JCT-VC, pero aumentando la complejidad del algoritmo y sus necesidades del hardware.

A pesar de las mejoras introducidas por el HEVC, la demanda global de contenido de vídeo en línea continuó incrementándose, lo cual demandaba la necesidad de una compresión aún más eficiente que la ofrecida por el estándar HEVC. Con este propósito, en 2015, la ITU-T VCEG y la ISO/IEC MPEG se unieron nuevamente formando el Joint Video Exploration Team (JVET).

Dos años más tarde, en 2017, sus actividades dieron como resultado el Joint Exploration Test Model (JEM), que demostró una reducción de tasa de bits superior al 30 % en comparación con HEVC. Dada esta evidencia, se decidió iniciar un nuevo esfuerzo de desarrollo estándar. El JVET pasó a llamarse Joint Video Experts Team, manteniendo el mismo acrónimo, y se emitió una convocatoria de propuestas en octubre de 2017. Esta convocatoria atrajo a más de 30 organizaciones para codificar tres categorías de contenido de vídeo: SDR, HDR y vídeo en 360º. Una evaluación subjetiva realizada en abril de 2018 demostró que todas las propuestas eran superiores a HEVC en términos de calidad subjetiva en la mayoría de los casos. Basándose en los elementos de tecnología de compresión más destacados de las propuestas, en abril de 2018 comenzó formalmente el proyecto para el desarrollo del estándar Versatile Video Coding (VVC), generando los primeros borradores del documento de especificación y el software para el modelo de prueba VVC (VTM) ese mismo mes. Finalmente, en julio de 2020 se publicó la primera versión del estándar VVC [55], [56], estando actualmente en fase de revisión la tercera revisión de la misma.

2.1.1. Fundamentos de la codificación de vídeo

Una secuencia de vídeo es una sucesión de imágenes que el ojo humano interpreta o percibe como movimiento. El proceso completo de transmisión de vídeo consta de las siguientes fases:

- Adquisición del vídeo a transmitir mediante captura analógica de secuencias de imágenes y digitalización de las mismas.
- Codificación y sub muestreo de crominancia de las muestras.
- Compresión del vídeo (con y/o sin pérdidas).
- Transmisión progresiva del vídeo comprimido.

Adquisición analógica de vídeo

Se capturan imágenes en dos dimensiones mediante lentes y un CCD (*Charge Coupled Device*) a intervalos regulares y se convierte en señales analógicas. Cada diodo del CCD recoge la intensidad lumínica para cada píxel de un color primario (RGB), haciendo uso de filtros para dejar pasar determinados colores. El número de píxeles que reciben luz verde es igual a la suma del número de píxeles que reciben luz roja y azul. La información de color que no se ha recibido en cada píxel es obtenida mediante interpolación de los píxeles adyacentes usando un DSP (*Digital Signal Processing*)

Digitalización

Se definen los parámetros de muestreo, cuantificación, barrido y resolución de imagen que se deben tomar para digitalizar una señal analógica.

Para la TV analógica existían principalmente dos formatos de barrido: NTSC, con 525 líneas y a una frecuencia de funcionamiento de 29.97 fotogramas por segundo; y PAL/SECAM, con 625 líneas y una frecuencia de 25 fotogramas por segundo.

Ambos estándares utilizan formato entrelazado, esto es, en cada barrido solo se muestra la mitad de las líneas de la imagen, por lo que es necesario dos fotogramas para obtener la imagen completa. Este sistema tiene la ventaja de que dobla la velocidad percibida de

la imagen, lo cual suaviza los movimientos y reduce efectos de parpadeo sin aumentar el ancho de banda disponible. En cambio, si la imagen contiene movimientos muy rápidos comienzan a detectarse artefactos o defectos en la imagen.

Codificación y sub muestreo

Cada muestra RGB se codifica con 24 bits/color, y posteriormente se realiza la conversión a YCrCb (YUV) mediante una matriz de conversión (Ecuación 2.1) para aumentar la compatibilidad con los sistemas monocromáticos:

$$\begin{pmatrix} Y' \\ U \\ V \end{pmatrix} = \begin{pmatrix} 0{,}229 & 0{,}587 & 0{,}114 \\ -0{,}147 & -0{,}289 & 0{,}436 \\ 0{,}615 & -0{,}515 & -0{,}100 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \tag{2.1}$$

Se realiza el sub-muestreo a las componentes de color Cr (U) y Cb (V) ya que el ojo humano percibe peor la información de color que la información del brillo, correspondiente a la componente de luminancia (Y). Cada uno de los componentes se codifica con una profundidad de 8 bits. En la Figura 2.1 se muestra una representación del proceso de conversión y sub-sampling explicado.

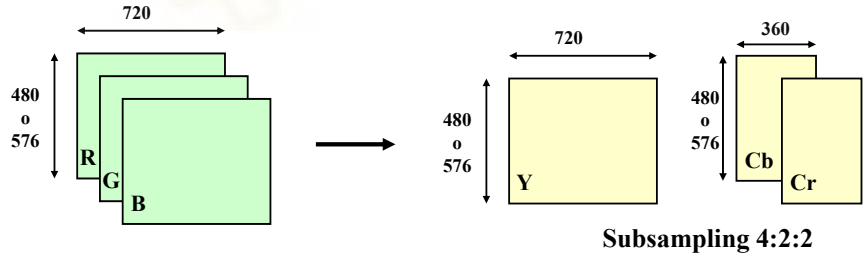


Figura 2.1: Conversión de RGB a YUV y sub-sampling de crominancia.

Compresión de vídeo en el dominio del píxel

Los algoritmos de compresión de vídeo se basan principalmente en localizar y minimizar la información redundante a lo largo de cualquier secuencia, sin perder la información necesaria para recuperar la secuencia. Para ello, se divide la imagen en pequeñas porciones, llamados macrobloques, y se trabaja de forma individual sobre cada uno para

elegir la mejor estrategia de compresión. La información redundante puede clasificarse principalmente en dos clases: espacial y temporal.

La redundancia espacial es aquella que está dentro de cada fotograma. Se basa en que los píxeles que están cerca unos de otros tienen un gran parecido entre sí que si los comparásemos con los píxeles de zonas más alejadas. Esto es debido a la naturaleza de las imágenes, que incluyen formas y objetos sólidos, con texturas uniformes. Teniendo en cuenta esto, es posible eliminar esta redundancia codificando la diferencia entre un píxel de referencia y el actual, también llamado residuo, y que generalmente tendrá un valor muy pequeño, lo cual reduce significativamente el número total de bits.

La redundancia temporal, también llamada *motion compensation*, viene dada por la similitud de cuadros sucesivos en una secuencia de vídeo. Se utilizan técnicas de codificación diferencial (DPCM), donde solamente se codificarán las diferencias entre cuadros sucesivos, que por norma general tendrán un tamaño de bits menor al que tendría el valor absoluto. La reconstrucción de un cuadro puede estar basado en otro(s) anterior(es).

Por ejemplo, el algoritmo MPEG-1 clasifica los cuadros en tres tipos, ya sea **I** (*Intracoded frames*), **P** (*Predictive frames*) o **B** (*Bidirectional frames*). Los fotogramas **I** codifican el cuadro de forma auto-contenida, es decir, sin utilizar referencias de otros cuadros. Los fotogramas **P** codifican el cuadro basándose en las diferencias respecto a un cuadro de referencia **I** anterior. Por último, los cuadros **B** se codifican mediante interpolación entre un cuadro anterior y otro posterior, de tipos **I** o **P**.

Debido a las propiedades de cada tipo de cuadro, los fotogramas de una secuencia de vídeo se agrupan en secuencias de cuadros (*Group Of Pictures*, GOP), que determinan el orden de forma que optimiza el nivel de compresión a la vez que asegura la reconstrucción correcta de la imagen en caso de errores. En la Figura 2.2 se muestra la secuencia utilizada por el estándar PAL (IBBPBBPBBI).

Cabe mencionar también el concepto de vectores de movimiento (*motion vector*), que consiste en localizar y determinar el desplazamiento de un macrobloque en el cuadro actual respecto a la posición que tenía en el cuadro de referencia. En la Figura 2.3 se muestra de forma gráfica el concepto de vector de movimiento.

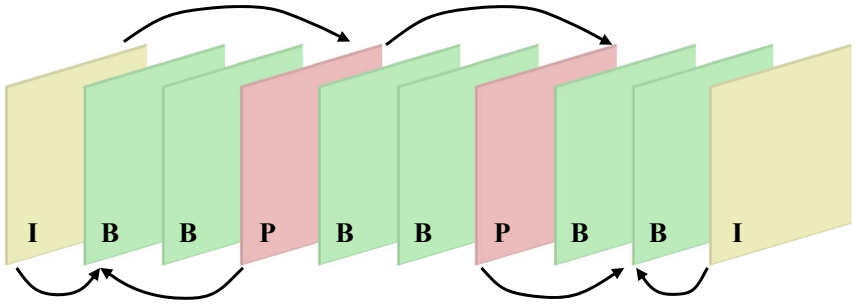


Figura 2.2: Secuencia de cuadros (GOP) para el algoritmo PAL (IBBPBBPBBI)

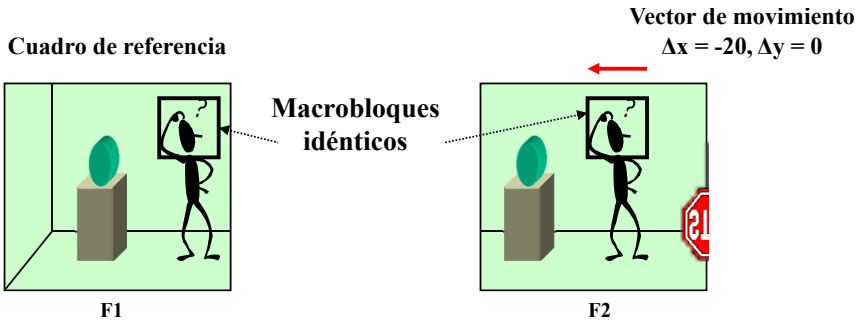


Figura 2.3: Representación gráfica del vector de movimiento para un macrobloque.

Compresión de vídeo en el dominio transformado

De forma análoga a como se realiza en el dominio del píxel, también es posible reducir información redundante en el dominio transformado. Esto se lleva a cabo transformando la señal al dominio frecuencial, generalmente utilizando la Transformada Discreta del Coseno (DCT), que compacta la mayor parte de la energía en la componente discreta (DC) y las bajas frecuencias. De esta forma, los coeficientes correspondientes a las altas frecuencias suelen tener un valor absoluto muy bajo o incluso nulo, lo cual permite despreciarlos sin que ello afecte de forma notable a la imagen reconstruida.

Posteriormente se puede aplicar una etapa de cuantificación ponderada para comprimir todavía más el resultado de la DCT. La ponderación consiste en aplicar a los coeficientes más importantes (DC y bajas frecuencias) una menor cuantificación que a los coeficientes menos importantes (medias y altas frecuencias), de esta forma el error producido no será perceptible. Cada algoritmo utiliza una matriz de cuantificación predeterminada, que se usa como divisor a los coeficientes transformados. Esta cuantificación puede tunearse en el proceso de codificación para obtener mayor o menor calidad de imagen.

Por último, se ordenan los coeficientes de la matriz en un vector siguiente determina-

dos esquemas de recorrido y se aplicarán técnicas de compresión estadística, mediante el uso de codificadores entrópicos.

2.2. Estándar de vídeo HEVC (H.265)

La mayoría de los estándares de codificación de vídeo previos al HEVC (H.261, H.262, MPEG-2, MPEG-4, H.264/AVC) se desarrollaron siguiendo la estructura llamada *Block-based Hybrid Video Coding* (Figura 2.4). De forma muy resumida esta estructura se basa en dividir la imagen en bloques para luego por cada bloque decidir el modo de codificación (Intra para tomar como referencia bloques vecinos del mismo *frame* o Inter para tomar como referencia bloques de otros *frames*, también llamado compensación de movimiento o *Motion Compensation*). El estándar HEVC también se ha desarrollado siguiendo esa estructura.

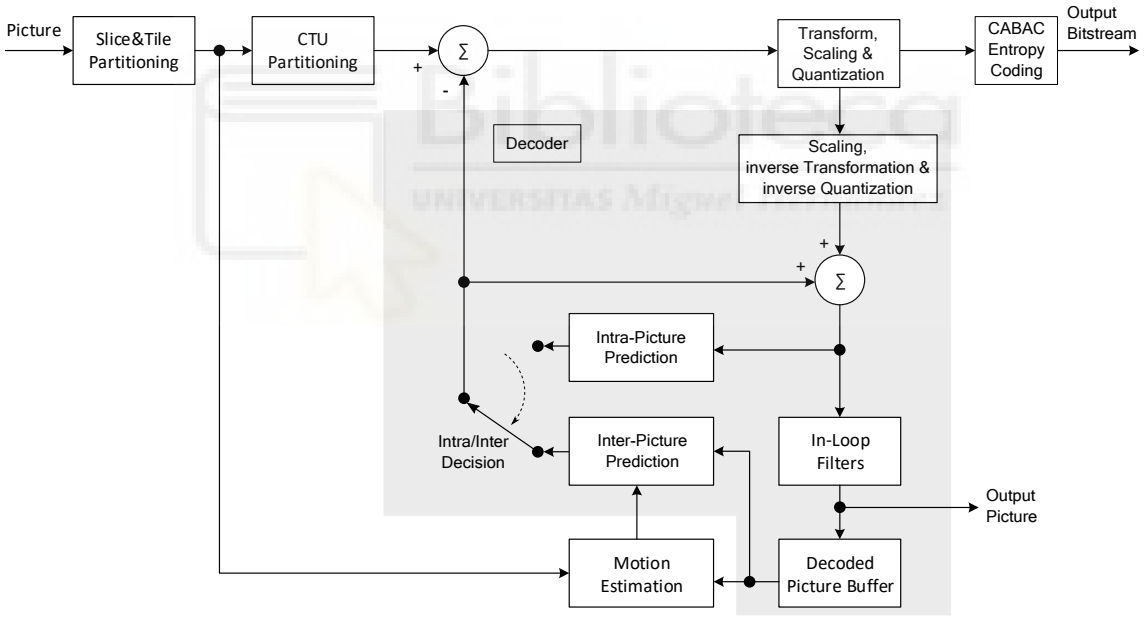


Figura 2.4: Diagrama de bloques de un codificador HEVC con decodificador incorporado (*región grisácea*).

La principal diferencia entre los estándares de vídeo mencionados es la cantidad y tipos de modos utilizados para los bloques de un *frame* de vídeo. Estos modos determinan primero si la predicción va a ser Intra o Inter, pero también determinan la subdivisión del "macrobloque" para las predicciones o transformaciones.

Para dar libertad a los fabricantes, el estándar de codificación HEVC solo especifica

la sintaxis del flujo de bits o *bitstream* y el resultado del proceso de codificación. Sin embargo, la eficiencia del codificador vendrá determinado en mayor medida por el algoritmo de codificación utilizado. Esto incluye la selección de modos de codificación, parámetros de predicción y cuantificación, así como los índices de cuantificación para los residuos en el dominio de la frecuencia.

Para optimizar el rendimiento del algoritmo de codificación se ha recurrido a sistemas basados en la optimización Rate-Distortion (RDO). Este método optimiza la cantidad de distorsión (pérdidas en la calidad del vídeo) frente al número total de bits a transmitir (la tasa de bits o *rate*).

Con este método se determinan los parámetros de codificación que minimizan una función de coste basada en la suma ponderada de la distorsión resultante, D, y el número de bits asociados, R, en el conjunto $\mathcal{A}$ de parámetros de codificación disponibles.

$$p^* = \arg \min_{\forall p \in \mathcal{A}} D(p) + \lambda \cdot R(p) \tag{2.2}$$

Donde el parámetro λ es una constante que determina el compromiso entre la distorsión (calidad de vídeo reconstruido) y el número o tasa de bits.

De todos los módulos de un codificador, el que realmente marca un salto de calidad en cuanto a la eficiencia de codificación es el aumento del número de modos soportados para codificar una imagen o bloque. Esto incluye aumento de precisión en los vectores de movimiento, amplia flexibilidad para elegir el orden de codificación de las imágenes, incremento del conjunto de imágenes de referencia, así como del número de modos para la predicción Intra, mayor flexibilidad en los tamaños de bloques de transformación, etc.

Dado un conjunto de bloques, diferentes divisiones en sub-bloques tienen asociados diferentes relaciones entre distorsión y tasa de bits. Cuando se divide un bloque en otros más pequeños generalmente se minimiza la distorsión, sin embargo, aumenta el número de bits necesarios para transmitir los parámetros de predicción.

Sin embargo, el aumento del conjunto de modos de subdivisión tiene el inconveniente de requerir equipos cada vez más potentes, por lo que también existe un compromiso entre la complejidad del codificador y la eficiencia computacional del mismo.

Durante el desarrollo del estándar HEVC se tuvo en cuenta el auge de los vídeos en

HD o Ultra-HD (4K y superior). Estas resoluciones de vídeo por lo general requieren un conjunto de tamaños de bloque elevados. En contra, tamaños de bloques elevados son ineficientes para resoluciones de vídeo bajas. Es por esto por lo que se ha implantado un sistema jerárquico de división de bloques, llamado Árbol cuaternario o *Quadtree-block Partitioning*, de forma que el codificador pueda rápidamente determinar el particionado óptimo a partir de la función de coste del R/D (Rate-Distortion).

2.2.1. Jerarquía de bloques

En la mayoría de estándares de codificación de vídeo previos al HEVC la imagen se particiona en bloques de 16×16 píxeles para la componente de luminancia o luma y en bloques de 8×8 píxeles para las componentes de crominancia o croma (para el sub muestreo de crominancia 4:2:0). Estos bloques reciben el nombre de macrobloques.

Por cada macrobloque se selecciona de forma independiente al resto un modo de codificación independiente, ya sea Intra o Inter. A su vez los macrobloques pueden dividirse en bloques más pequeños con el objetivo de reducir la distorsión. Aunque la mayoría de los estándares previos siguen usándose en la actualidad, estos se diseñaron pensando en secuencias de vídeo con resoluciones que actualmente denominaríamos medias o bajas, por lo que su uso en secuencias con resoluciones elevadas como HD o 4K, muy populares en la actualidad, es muy ineficiente debido a la restricción del tamaño de los macrobloques. Es por esto por lo que uno de los aspectos en el diseño del particionado en el HEVC fue incrementar y flexibilizar el tamaño de los "macrobloques".

En el estándar HEVC, cada imagen se divide en bloques de tamaño $2^N \times 2^N$ para la componente de luminancia y en bloques de $2^{N-1} \times 2^{N-1}$ para ambas componentes de crominancia llamados *Coding Tree Block* o CTBs. El conjunto de los tres CTB junto con la sintaxis asociada forma una entidad llamada *Coding Tree Unit* o CTU, y representa la unidad básica de procesamiento en el estándar HEVC. El tamaño N es elegido por el codificador entre los valores $N = 4, 5$ y 6, dando lugar a bloques de tamaño 16×16, 32×32 y 64×64 respectivamente (para luma). El codificador elegirá el valor de N en función del hardware disponible, el archivo fuente a codificar, el R/D de la codificación, entre otros.

En los estándares previos al HEVC el macrobloque no solo se utiliza para particionar

la imagen, sino que representa la unidad de procesamiento por el cual el codificador elige el modo de codificación. A su vez, dependiendo del modo seleccionado, el macrobloque de tamaño 16×16 se puede dividir en bloques más pequeños con el fin de mejorar el coste Rate-Distortion, tal y como muestra la Figura 2.5.

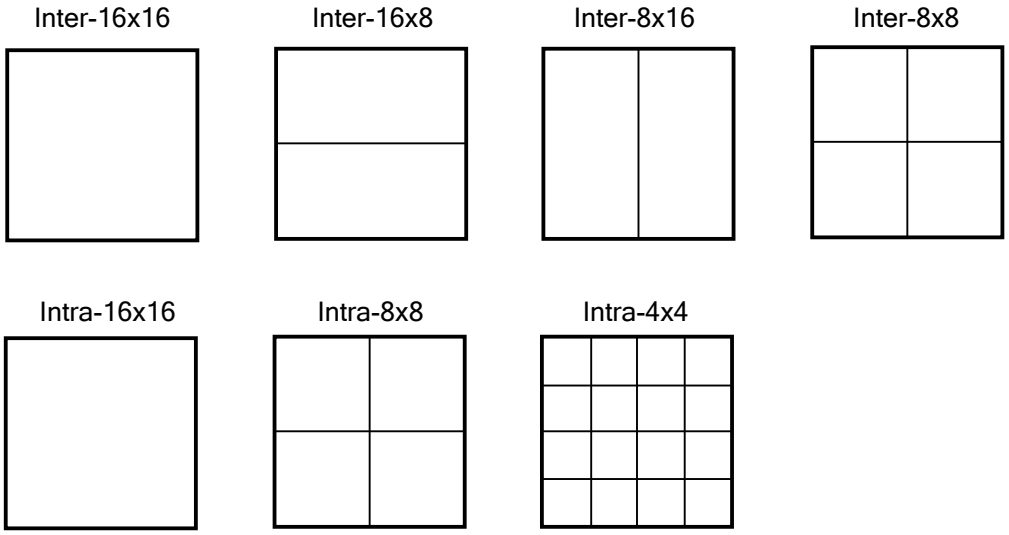


Figura 2.5: Modos de particionado de macrobloques en el estándar H.264/AVC para modos de codificación Inter (*arriba*) e Intra (*abajo*) [57].

En HEVC se ha buscado flexibilizar al máximo el tamaño de los bloques con el fin de optimizar la elección del modo de codificación y aumentar la eficiencia del codificador. Por ello se ha diseñado una nueva unidad de procesamiento llamado *Coding Unit* o CU. Cada CTU puede ser dividido en diferentes CUs de tamaño variable. Es por ello por lo que cada CTU tiene asociada una sintaxis donde se especifica el particionado del mismo, llamado *Coding Tree* o CT.

Al igual que los CTUs, un CU es una entidad que incluye un *Coding Block* o CB de luminancia, dos CBs de crominancia y una sintaxis asociada. En una configuración típica, el rango de tamaños de CB se mueve desde 8×8 hasta 64×64 (para luminancia).

Los CUs de cada CTU son recorridos y codificados siguiendo un algoritmo de búsqueda en profundidad (del inglés Depth-first Order o DFO) de forma que, exceptuando los bloques frontera con la parte superior y el lateral izquierdo, los CUs a codificar tienen sus CUs vecinos (el CU superior y el izquierdo) codificados, por lo que pondrán ser utilizados como referencia en el modo Intra. Este orden de codificación es también conocido como Z-Scan.

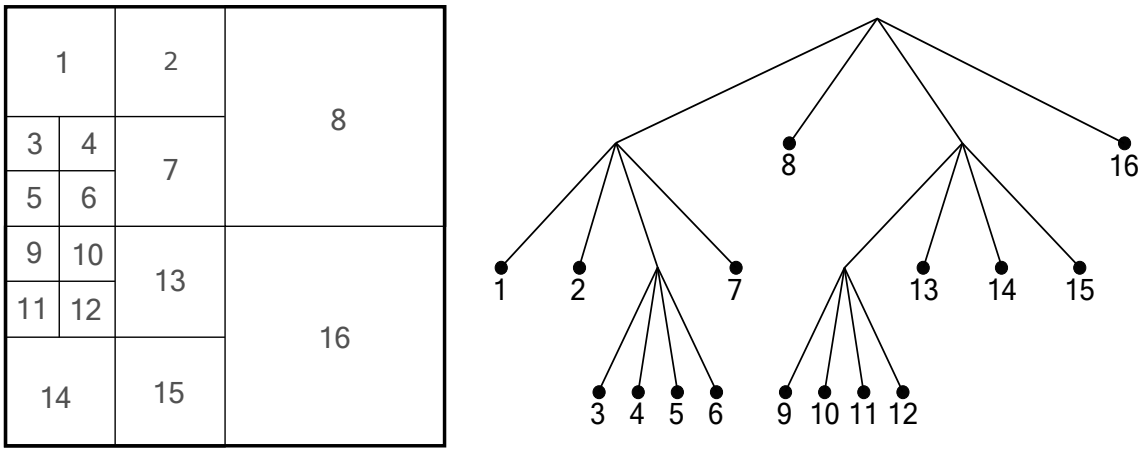


Figura 2.6: Ejemplo de particionado de un CTUs de 64×64 en diferentes CU, siguiendo el orden de codificación según el algoritmo *quadtree* [57].

El tamaño de la imagen a codificar (anchura y altura) debe ser un entero múltiplo del tamaño mínimo del CU, pero no es necesario que sea múltiplo del tamaño de los CTUs, en cuyo caso los CTUs frontera afectados se dividirán hasta que el tamaño coincida con los bordes. Los CUs localizados fuera de la imagen no serán codificados.

Los CU representan la unidad de procesamiento por la cual el codificador elige el modo de codificación (Intra o Inter), de la misma forma que ocurre con los macrobloques en los estándares previos al HEVC. La principal ventaja de este último radica en que el tamaño de las unidades de procesamiento es variable, y la introducción de la estructura *quadtree* permite obtener una sintaxis elegante y unificada para especificar el particionado de los CTUs.

A cada CU se le asigna un modo de codificación, que puede ser Intra o Inter. Si es Intra, se selecciona una de las 35 predicciones espaciales para la componente de luminancia. Si el tamaño del CU es el mínimo configurado, el CB de la componente de luminancia se puede dividir en cuatro sub-bloques, teniendo cada uno una predicción independiente. Sin embargo, esto no ocurre con los CBs de crominancia debido al bajo impacto que tiene la codificación croma en la eficiencia del codificador. Para el caso de las componentes de crominancia la predicción espacial se elige de entre 5 candidatos, siendo uno de ellos el candidato escogido para luminancia. Una vez el CU tiene escogida una predicción espacial esta no siempre se aplica a todo el bloque. Un CB puede dividirse en múltiples *Transform Block* o TB, que representan la unidad a la cual se realizará la transformada 2D al residuo. El particionado se realiza siguiendo el algoritmo *quadtree*, donde el CU es la

raíz del árbol (llamado en este caso *Residual Quadtree Transform* o RQT). Esta división de CB en TB mejora la eficiencia del codificador, ya que logra separar zonas de la imagen con distinto valor frecuencial, mejorando así el coste Rate-Distortion.

Dentro de un CB particionado en diferentes TB se aplica la misma predicción espacial, sin embargo, es posible que la referencia la tome de un TB adyacente en lugar de de otro CB vecino, mejorando todavía más el rendimiento del codificador (Fig. 2.7)

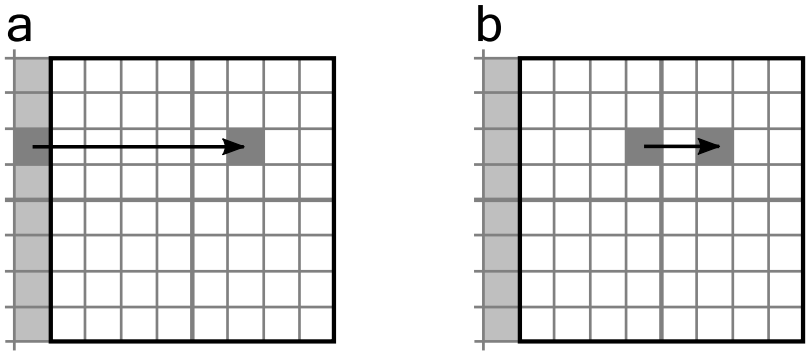


Figura 2.7: Ilustración de codificación Intra horizontal de un *Coding Block* (CB) de 8×8 divido en cuatro *Transform Block* (TB), obteniendo la referencia de un CB adyacente (**a**) o de un TB adyacente (**b**) [57].

Por otro lado, si el CU es codificado en modo Inter, los bloques de luminancia y crominancia pueden dividirse en entidades llamadas Prediction Block o PB. Un PB es un bloque que usa los mismos parámetros de movimiento para las predicciones, que incluyen el número de hipótesis de movimiento, así como los índices de las imágenes de referencia y los vectores de movimiento asociados a cada hipótesis. El PB de la componente de luminancia y las componentes de crominancia, así como la sintaxis asociada forman una entidad llamada Prediction Unit o PU. HEVC soporta múltiples modos de particionado de CUs en PUs, lo cual mejora considerablemente la eficiencia del codificador. En la Tabla 2.1 se muestran los tamaños de bloque disponibles para la codificación Inter de diferentes estándares de vídeo.

Sin embargo, esta mejora solo puede ser explotada si el codificador evalúa un gran número de particiones distintas, de lo contrario el sobrecoste en la señalización hará que aumente la tasa de bits, disminuyendo así la mejora obtenida. Es por ello por lo que el codificador puede configurarse para evaluar un conjunto reducido de particiones, enviando esta información en la señalización inicial de la transmisión, reduciendo así el sobrecoste

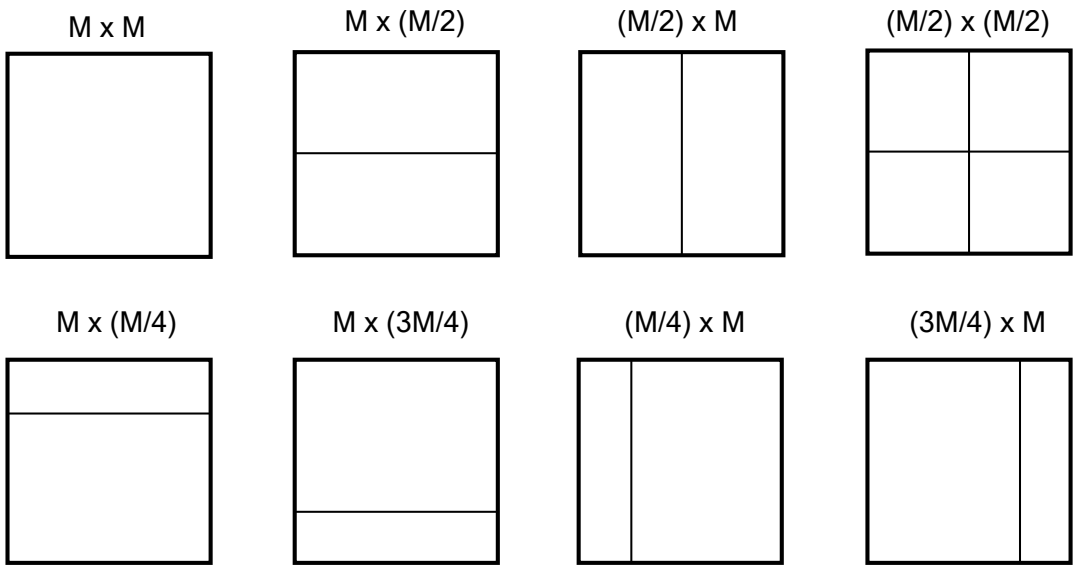


Figura 2.8: Modos de particionado de CUs en Prediction Units (PU). Dependiendo del tamaño de CU algunos modos no estarán soportados [57].

y mejorando finalmente la eficiencia en codificadores simples o que no disponen de la potencia o memoria suficiente para explotar todas las posibilidades.

Tabla 2.1: Comparación de tamaños de bloque soportados para la predicción Inter en diferentes estándares de vídeo.

Estándares de vídeo	Tamaños de bloque soportados para la predicción Inter
H.262 | MPEG-2 Video	16×16
H.263	16×16, 8×8
MPEG-4 Visual	16×16, 8×8
H.264 | MPEG-4 AVC	16×16, 16×8, 8×16,8×8,8×4,4×8,4×4
H.265 | HEVC	64×64, 64×48, 64×32, 64×16, 48×64, 32×64, 16×64, 32×32, 32×24, 32×16, 32×8, 24×32, 16×32, 8×32, 16×16, 16×12, 16×8, 16×4, 12×16, 8×16, 4×16, 8×8, 8×4, 4×8

2.2.2. Predicción Intra en HEVC

En comparación con el estándar previo H.264 que contiene 8 predicciones angulares más la DC, HEVC incrementa el número de predicciones hasta los 35 modos, de los

cuales 33 son angulares y los otros dos corresponden a los modos DC y planar.

Todas las predicciones Intra usan como referencia los bloques reconstruidos adyacentes y dado que los bloques se reconstruyen al nivel de los *Transform Block* (TB) la predicción Intra también se realiza a este mismo nivel, siendo las dimensiones del bloque de tamaño variable.

HEVC permite para todos los tamaños de bloque calcular cada uno de los 35 modos de predicción, por lo que en el proceso de diseño se desarrolló un algoritmo capaz de realizar los cálculos de la forma más sencilla computacionalmente. Además, dependiendo del tamaño del bloque, el modo de predicción y la direccionalidad, el codificador puede aplicar un filtro de preprocesado a las muestras de referencia.

También incorpora un filtro de postprocesamiento, cuyo uso se hace notable en los modos de predicción vertical, horizontal y DC (ver Figura 2.9). Este filtro ayuda a reducir el efecto de blocking.

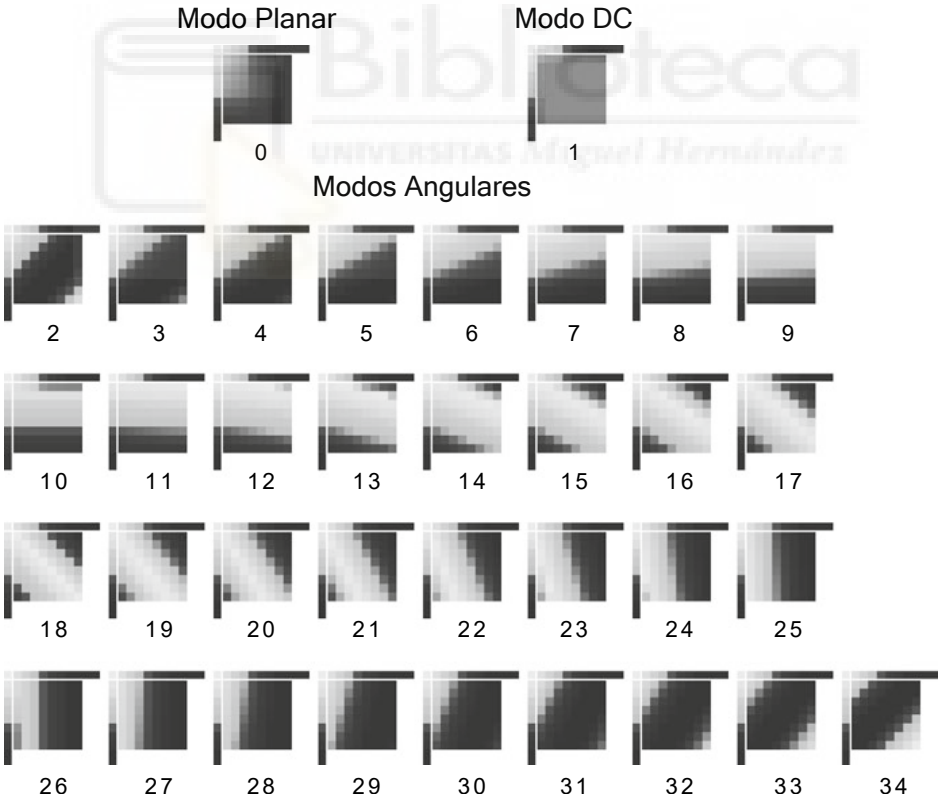


Figura 2.9: Ejemplo de Prediction Blocks (PB) de tamaño 8×8 generados con todas las predicciones de HEVC [57]

Como ya se ha mencionado anteriormente, la predicción Intra en HEVC se realiza a

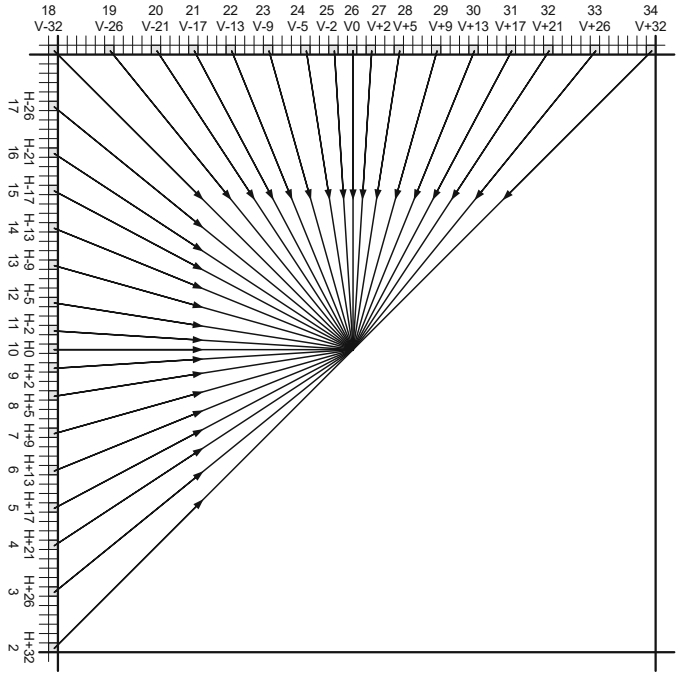


Figura 2.10: Representación gráfica del set de 33 modos de predicción angulares en el estándar HEVC. Los modos pueden clasificarse según sean horizontales (2-17) o verticales (18-34), así como positivos (2-10, 26-34) o negativos (11-25) [57]

partir de los valores de los bloques adyacentes reconstruidos. Sin embargo, en ocasiones algunos de estos bloques no están disponibles parcial o totalmente, por lo que el codificador se encarga de rellenar estos huecos con el fin de poder usar todo el conjunto de predicciones. En el caso extremo en que no exista ningún bloque adyacente disponible el codificador rellenará los huecos con el valor promedio de luminancia (que para una profundidad de 8 bits corresponde un valor de 128), mientras que si solo hay un único bloque disponible el resto tomará su valor. En caso de tener numerosos huecos el codificador irá completándolos, copiando el mismo valor de los bloques adyacentes en el sentido de las agujas del reloj.

Una vez obtenidas las referencias es posible que se aplique un filtro postprocesado con el fin de mejorar las predicciones reduciendo artefactos no deseados. La elección de usar o no este filtro la toma el codificador en base al tamaño del bloque y el modo de predicción (por ejemplo, en bloques de 4×4 o modo DC no aplica). Si se decide usar el filtrado uno de los dos algoritmos disponibles será utilizado. Por defecto se utilizará el filtro de suavizado 3-Taps, donde a excepción de los píxeles situados en los extremos el resto son filtrados en función de los píxeles vecinos [1 2 1]/4 (Fig. 2.11 **a**). El otro algoritmo modifica el

valor de los píxeles de referencia mediante interpolación lineal entre los tres píxeles de referencia extremos (Fig. 2.11 **b**).

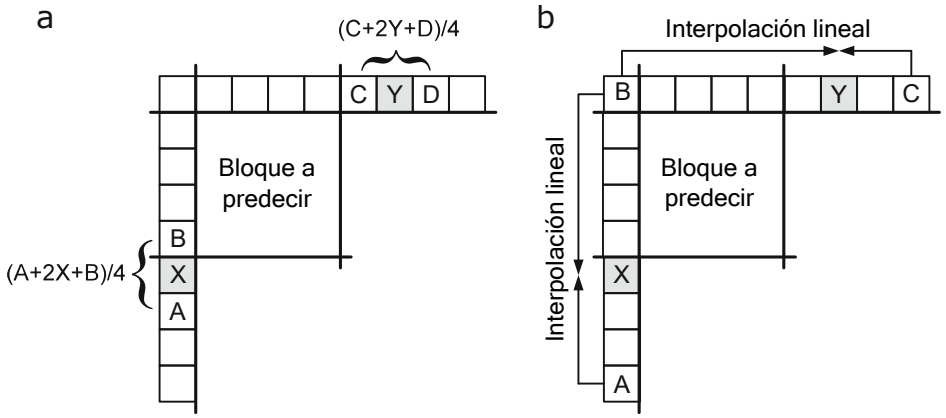


Figura 2.11: Métodos de postprocesado de muestras de referencia: (**a**) filtro 3-Taps, (**b**) interpolación lineal [57]

Este algoritmo modifica prácticamente todos los píxeles de referencia y es útil para evitar artefactos o efectos de blocking en imágenes extremadamente suaves.

Predicción Angular, DC y Planar de muestras Intra

El estándar HEVC incrementa el número de predicciones de muestra con respecto a estándares anteriores hasta alcanzar un total de 35 métodos de los cuales 33 son predicciones angulares, que permiten modelar las estructuras direccionales típicas presentes en las imágenes, y otros dos métodos (DC y Planar) útiles en zonas lisas o con texturas complejas. Aun teniendo un gran número de predicciones, el estándar se ha desarrollado de forma que el cálculo de las diferentes predicciones tenga un coste computacional reducido, haciendo uso de técnicas como la extensión de fila (o columna) de referencia, aplicable a las predicciones angulares negativas.

Esta técnica busca tener un único vector de referencia, en lugar de tener dos vectores (uno columna y otra fila). Dependiendo del modo de predicción angular seleccionado la creación del vector unidimensional de referencia se realizará copiando las muestras desde la dirección de predicción (modos angulares positivos) o mediante la proyección de muestras de la columna de referencia si se está extendiendo la fila, o la proyección de muestras de la fila de referencia si se está extendiendo la columna (modos angulares negativos). En la Figura 2.12 se puede observar el uso del método de extensión de muestras

para completar el vector unidimensional.

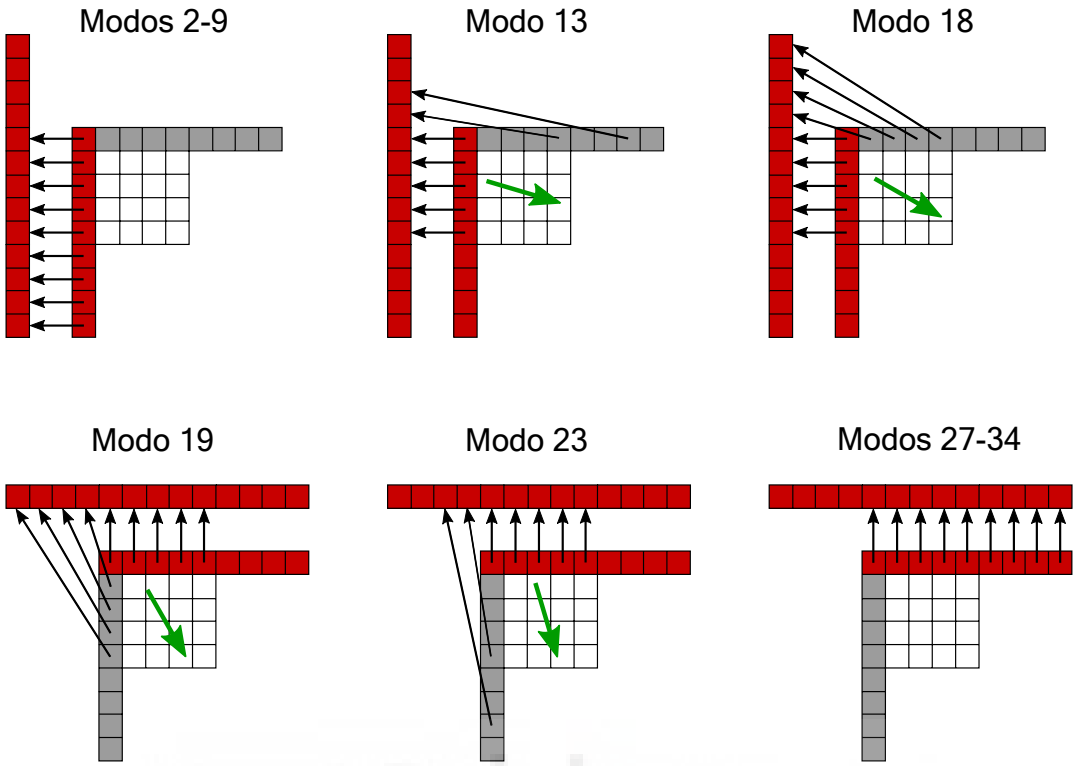


Figura 2.12: Representación gráfica de la creación de vectores de referencia unidimensionales en algunos modos angulares positivos y negativos

Una vez formado el vector unidimensional de referencia se calcula el valor de cada píxel del bloque a predecir a partir de una expresión matemática que tiene en cuenta el modo angular elegido y la posición del $[x][y]$ del píxel.

La predicción DC se calcula tomando el valor promedio de los bloques adyacentes superior y lateral izquierdo mientras que la predicción Planar se lleva a cabo promediando las predicciones lineales horizontales y verticales.

El modo DC y los modos angulares 10 y 26 (horizontal y vertical puros) tienen, además, dependiendo del tamaño de bloque, una etapa de postfiltrado.

Elección de modo de codificación Intra

A diferencia del estándar H.264/AVC donde únicamente se elige el mejor modo de codificación, en el estándar HEVC se eligen los tres mejores candidatos (MPM) debido a que elegir un único candidato resultó ineficiente teniendo en cuenta el incremento del número de modos posibles.

La elección de los tres modos más probables de predicción se basa en un simple algoritmo que tiene en cuenta en modo escogido en los bloques adyacentes superior y lateral izquierdo.

2.2.3. Transformación, escala y cuantificación

En la estructura *Block-based Hybrid Video Coding*, usada por el estándar HEVC, así como alguno de sus predecesores la transformación se aplica al error residual de predicción, de esta forma se eliminan sus componentes frecuenciales elevadas. Cuanto mayor sea el tamaño del bloque, mejor compactará la energía. Como vimos en el apartado 2.2.1, la transformada se realiza al nivel jerárquico de los *Transform Block* (TB) que pueden tener un tamaño de 4×4, 8×8, 16×16 y 32×32.

La transformada que más se utiliza a la hora de codificar imágenes es la transformada discreta de coseno (DCT) que está basada en la transformada discreta de Fourier (DFT) pero a diferencia de esta última la DCT compacta la mayor parte de la energía en los coeficientes más cercanos a la DC. También se diferencia en que sus componentes son números reales enteros. Los N coeficientes de transformación v_i aplicados a una señal de entrada u_j puede expresarse como

$$v_i = \sum_{j=0}^{N-1} u_j \, c_{ij} \,, \qquad (i = 0, \cdots, N-1) \tag{2.3}$$

donde el coeficiente c_{ij} se define como

$$c_{ij} = \frac{P}{\sqrt{N}} \cos\left(\frac{(2j+1)i\pi}{2N}\right), \qquad (i, j = 0, 1, \cdots, N-1), \quad P = \begin{cases} \sqrt{2} & i = 0 \\ 1 & i > 0 \end{cases} \tag{2.4}$$

Se define también los vectores base c_i de la DCT como $c_i = [c_{i0}, \cdots, c_{i(N-1)}]^T$ donde $i = 0, \cdots, N-1$.

Para realizar la transformada se necesita la función bidimensional de la DCT (DCT 2-D), que puede obtenerse como producto de dos DCT 1-D; el primero corresponde al eje horizontal y el segundo al vertical. Las funciones base de la DCT 2-D se muestran en la Figura 2.13.

En la mayoría de los estándares de vídeo, el algoritmo para obtener la transformada resulta del producto de tres matrices 2-D: dos de ellas corresponden a la matriz de transformación base de la DCT

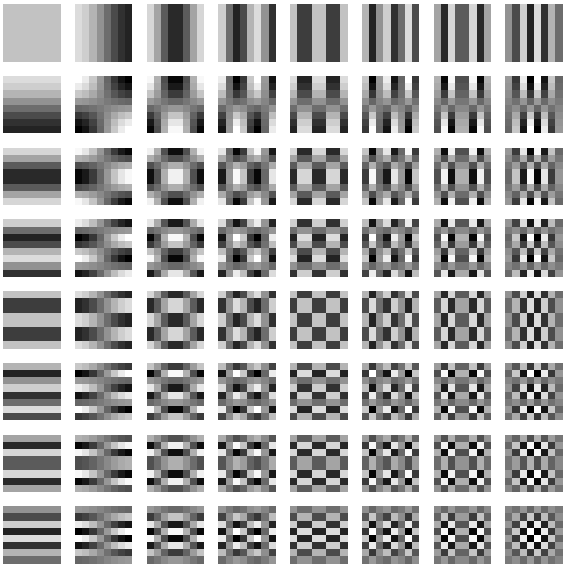


Figura 2.13: Funciones base de la DCT 2D de un bloque de 8×8.

Debido a las propiedades de la matriz de coeficientes de transformación así como los diferentes tamaños de bloques disponibles, el cálculo de la DCT y su inversa en el estándar HEVC se descompone en cálculos más pequeños utilizando estructuras parciales de butterfly, de esta forma la complejidad de los cálculos se reduce y la implementación del código en software como hardware se simplifica [58].

HEVC aplica una aproximación finita de la DCT para cada uno de los tamaños de TB posibles. Además para algunos modos angulares de predicciones Intra de tamaño 4×4 se especifica una transformada discreta de seno (DST), ya que se ha comprobado que compacta mejor la energía en estos casos [59].

El motivo de utilizar una aproximación finita ha sido el de simplificar la algoritmia para reducir el número de operaciones matemáticas y por tanto el tiempo necesario para su cómputo. Esta aproximación a enteros comenzó a utilizarse en el estándar H.264/AVC, sin embargo en HEVC se han escalado[1] las matrices de transformación para que los vectores base de cada tamaño de matriz tenga la misma energía, reduciendo todavía más la complejidad ya que se evita tener que compensar las diferentes normalizaciones tal y como ocurre en H.264/AVC. Esta característica hace que a partir de la definición de la matriz de transformación 32×32 se pueda obtener por sub muestreo el resto de las matrices. A

[1]HEVC utiliza la siguiente escala para las matrices de transformación DCT: $2^{(6+M/2)}$, donde $M = log_2(N)$ y N es el tamaño de la matriz cuadrada

modo de ejemplo se muestran en la Figura 2.14 las matrices de transformación de la DCT 2-D para tamaños de bloque 8×8 y 4×4, así como la matriz de transformación de la DST que incorpora el estándar HEVC.

DCT 8×8

$$\begin{pmatrix} 64 & 64 & 64 & 64 & 64 & 64 & 64 & 64 \\ 89 & 75 & 50 & 18 & -18 & -50 & -75 & -89 \\ 83 & 36 & -36 & -83 & -83 & -36 & 36 & 83 \\ 75 & -18 & -89 & -50 & 50 & 89 & 18 & -75 \\ 64 & -64 & -64 & 64 & 64 & -64 & -64 & 64 \\ 50 & -89 & 18 & 75 & -75 & -18 & 89 & -50 \\ 36 & -83 & 83 & -36 & 36 & 83 & -83 & 36 \\ 18 & -50 & 75 & -89 & 89 & -75 & 50 & -18 \end{pmatrix}$$

DCT 4×4

$$\begin{pmatrix} 64 & 64 & 64 & 64 \\ 83 & 36 & -36 & -83 \\ 64 & -64 & -64 & 64 \\ 36 & -83 & 83 & -36 \end{pmatrix}$$

DST 4×4

$$\begin{pmatrix} 29 & 55 & 74 & 84 \\ 74 & 74 & 0 & -74 \\ 84 & -29 & -74 & 55 \\ 55 & 85 & 74 & -29 \end{pmatrix}$$

Figura 2.14: Matrices de transformación para DCT 8×8, DCT 4×4 y DST 4×4 en HEVC.

HEVC utiliza el mismo esquema de cuantificación lineal, y de reconstrucción por escalado uniforme utilizado por H.264/AVC, con 52 niveles determinados por el parámetro QP, $QP = 0, \cdots, 51$, el cual duplica su tamaño de cuantificación cada 6 niveles, permitiendo un control logarítmico de la cuantificación [60]. De modo genérico, el proceso de cuantificación se obtiene aplicando la expresión matemática descrita en la Ecuación 2.5, donde $C(x, y)$ es la matriz de coeficientes transformados, $F(qp)$ es el vector con los factores de cuantificación definidos en la Ecuación 2.6, N es el tamaño del TB y $C_{qp}(x, y)$ es la matriz de coeficientes cuantificados:

$$C_{qp}(x,y) = \left(\frac{C(x,y) * F[QP\,\%6]}{2^{(29+\frac{QP}{6}-N-BitDepth)}}\right) \tag{2.5}$$

$$F(qp) = [26216, 23302, 20560, 18396, 16384, 14564] \tag{2.6}$$

Al igual que el estándar H.264/AVC, al codificador HEVC se le puede indicar si usar una matriz escalar de cuantificación uniforme (*flat*) o no-uniforme. La matriz no-uniforme cuantifica los coeficientes dependiendo de su posición. Este tipo de cuantificación se basa en los estudios del Sistema Humano Visual (HVS) [61], los cuales demuestran que

las componentes frecuenciales muy altas pueden cuantificarse en mayor medida ya que, perceptualmente, no se notarán las diferencias (ver Apartado 2.3.1).

Para reducir la memoria necesaria para almacenar valores de escalado específicos de la frecuencia, el estándar HEVC únicamente define dos matrices de cuantificación no-uniformes de tamaño 8×8, una para la codificación intra y otra para la codificación inter, que serán aplicadas al canal de luminancia y los dos canales cromáticos. Para tamaños de bloque 4×4, no se aplica este tipo cuantificación, mientras que, para tamaños de bloques de 16×16 y 32×32 píxeles, las matrices se construyen a partir de la matriz de tamaño 8×8 por sobre muestreo, replicando los valores tal y como vemos en la Figura 2.15.

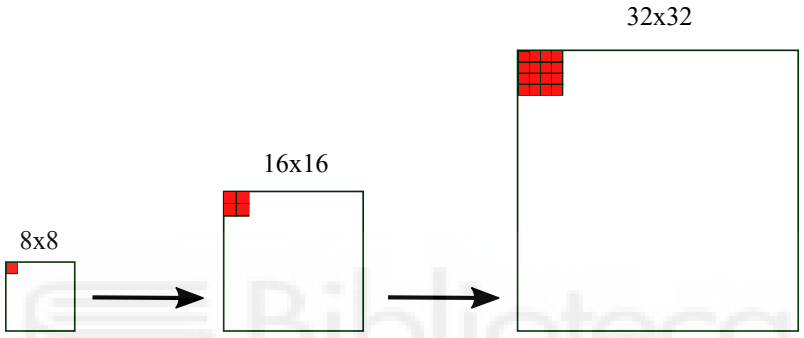


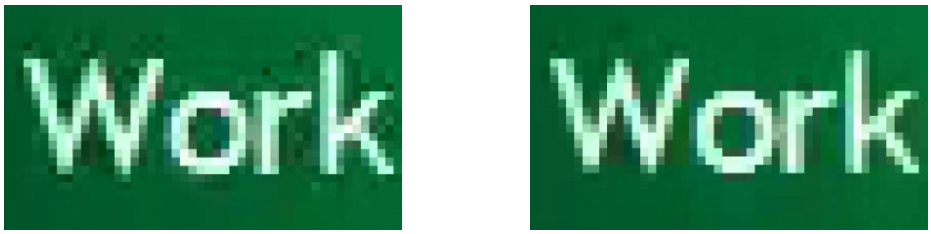
Figura 2.15: Construcción de matrices de cuantificación para TB de tamaño 16×16 y 32×32 a partir de la matriz de cuantificación de tamaño 8×8 [57]

2.2.4. Filtros In-loop

El estándar HEVC define dos tipos de filtros *in-loop*, el filtro de *deblocking* (DB) y el filtro *Sample Adaptative Offset* (SAO). Los filtros *in-loop* se utilizan durante el proceso de codificación y decodificación, después de realizar la cuantificación inversa y antes de guardar la imagen en el búfer de imágenes decodificadas.

Primero se aplica el filtro de *deblocking*, diseñado para atenuar las discontinuidades en los bordes de los *Prediction Units* (Inter) y *Transform Units* (Intra) de tamaño 8×8 o superior. Después se aplica el filtro SAO, cuyo objetivo es, entre otros, el de corregir defectos de codificación, como los artefactos de anillo o el "*banding*" en zonas grandes. En la Figura 2.16 se puede observar cómo este filtro corrige estos defectos en una imagen de muestra.

La ventaja más importante de los filtros *in-loop* es la de mejorar la calidad perceptual



(a) (b)

Figura 2.16: Ejemplo de aplicación de filtro SAO: (a) filtro SAO desactivado, (b) filtro SAO activado [62]

de las imágenes reconstruidas, además de incrementar la calidad de las imágenes de referencia y, por tanto, la eficiencia de compresión. Más adelante, en las secciones 3.1.2 y 3.1.3, se detalla en profundidad el algoritmo utilizado por estos filtros.

2.2.5. Codificación entrópica

La codificación entrópica es una técnica de compresión sin pérdidas que utiliza propiedades estadísticas de modo que el número de bits usados para representar los datos sea logarítmicamente proporcional a la probabilidad de los datos, es decir, a mayor probabilidad de aparición, los datos se codificarán con una longitud mínima, mientras que los datos poco frecuentes se codificarán con más bits.

El estándar HEVC utilizan el codificador entrópico CABAC (*Context-Base Adaptative Binary Arithmetic Coding*), que se emplea como último paso del proceso de codificación. Este codificador es uno de los tres codificadores entrópicos definidos en el estándar H.264/AVC, sin embargo, durante el desarrollo del HEVC y debido a los buenos resultados en cuanto a la eficiencia de compresión se determinó que sería el único método elegible para este estándar [57].

El algoritmo CABAC consta de tres etapas principales, que son la binarización, el modelado de contexto y la codificación binaria aritmética. La binarización mapea los elementos de la sintaxis a símbolos binarios, también llamados *bins*. El modelado de contexto estima la probabilidad de cada *bin* basándose en un determinado contexto. Finalmente, la codificación binaria aritmética (BAC) comprime los *bins* a bits de acuerdo con la probabilidad estimada.

2.3. Codificación perceptual

En un esquema de codificación no adaptativo, los coeficientes son cuantificados usando un valor fijo, para después pasarlos por un codificador entrópico y así reducir la redundancia.

Muchos codificadores, con independencia del tipo de transformada utilizada, determinan un valor límite (umbral) por el cual los coeficientes que se encuentran por debajo se establecen a cero. El objetivo es tener un balance entre la pérdida de calidad debido a la cuantificación y la reducción de la tasa de bits. Cuanto mayor sea el número de coeficientes reducidos a cero, más deteriorada estará la imagen reconstruida. Sin embargo, el modo en que la imagen se deteriora no solo depende del número de coeficientes reducidos a cero, sino de qué coeficientes son los descartados, ya que perceptualmente algunos son más importantes que otros.

Otro factor importante consiste en determinar el nivel de cuantificación, ya que sobre cuantificar los coeficientes de baja frecuencia de la DCT, provoca efectos de blocking, mientras que sobre cuantificar los coeficientes de altas frecuencias puede hacer que se haga visible el ruido.

Adicionalmente, existen diferentes fenómenos de enmascaramiento (*masking*) que pueden tenerse en consideración para cuantificar determinadas zonas de la imagen manteniendo la misma calidad perceptual, ya que explotan las características del Sistema Visual Humano (HVS). A continuación, se detallan algunos de estos fenómenos.

2.3.1. Contrast Sensitivity Function (CSF)

El HVS es capaz de percibir pequeñas diferencias de luminosidad [63]. Sin embargo, esta mínima diferencia varía en función de la luminosidad del fondo de la imagen. Esta dependencia a la luminosidad del fondo es lo que llamamos sensibilidad al contraste (*contrast sensitivity*). Un modelo básico para esta dependencia es la ley de Weber-Fechner. Esta ley establece que el menor cambio discernible en la magnitud de un estímulo es proporcional a la magnitud del estímulo. Matemáticamente, el contraste de Weber puede expresarse como la Ecuación 2.7.

$$C^W = \frac{\Delta L}{L} \tag{2.7}$$

La ley de Weber-Fechner no satisface todos los niveles de fondo de luminosidad. Solo funciona para niveles entre 10^{-1} cd/m^2 y 10^3 cd/m^2; fuera de este rango el umbral de contraste se incrementa, es decir, hay menos sensibilidad al contraste. Evidentemente, la ley de Weber-Fechner es solo una aproximación matemática de la percepción sensorial, pero se ha utilizado ampliamente en estudios científicos para realizar las mediciones de contraste.

El contraste es la diferencia relativa de nivel de luminosidad de dos puntos adyacentes de una imagen o campo visual. Esto es, el contraste es la diferencia de luminosidad o color que hace a un objeto distinguible. El HVS es más sensible a los cambios de luminosidad (contraste) que, a la luminosidad absoluta, por lo que podemos percibir objetos a pesar de los cambios de iluminación (por encima de 10^{-1} cd/m^2, tal y como establece la ley de Weber-Fechner) siempre que el contraste sea suficientemente alto.

Si el contraste es muy bajo no podremos distinguir un objeto del fondo. En estos casos, algunos objetos de la escena se vuelven invisibles. Se dice que estos objetos se encuentran por debajo del umbral de contraste.

La sensibilidad es inversamente proporcional al umbral de contraste, esto es, cuanto más bajo es el contraste que necesitamos para percibir un objeto de una escena, más alta es nuestra sensibilidad, y viceversa. Bajo condiciones óptimas, el umbral de contraste puede ser inferior al 1 %.

En la Figura 2.17 podemos ver tres imágenes de rejillas senoidales, donde cambian gradualmente la luminancia sobre el eje X, cada una con una diferencia entre la luminosidad máxima y mínima diferente. Debajo de las imágenes podemos ver la onda senoidal que representa ese cambio de niveles.

El contraste de un estímulo periódico (habitualmente senoidal) con diferentes frecuencias se conoce como contraste de Michelson [64], y se define con la fórmula $C_M = \frac{L_{\max} - L_{\min}}{L_{\max} + L_{\min}}$, donde $L_{\max}$ y $L_{\min}$ son la luminancia máxima y mínima respectivamente. Si la onda senoidal fuese una línea horizontal, $L_{\max}$ y $L_{\min}$ valdrían lo mismo, con lo que no habría contraste: $C_M = 0$; en cambio, si la amplitud de la onda senoidal fuese lo suficientemente

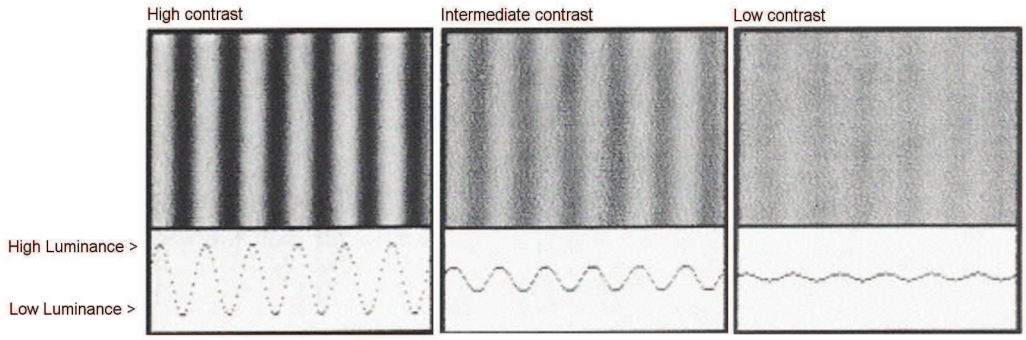


Figura 2.17: Tres rejillas senoidales con la misma frecuencia espacial pero con diferente contraste (descendente de izquierda a derecha)

elevada, el valor máximo de contraste resultaría en $C_M = 1$.

Pero, si en las imágenes anteriores hubiese más de un objeto y esos objetos tuviesen diferente tamaño, forma y textura, entonces el punto en el que cada objeto se vuelve invisible sería diferente. Esto es debido a que la percepción humana del contraste no solo depende de la diferencia de luminosidad sino también de la frecuencia espacial. Por tanto, el umbral de contraste varía con la frecuencia espacial.

Supongamos que usamos una lente para capturar la imagen de una rejilla senoidal en un folio blanco. Esta rejilla tiene unas propiedades físicas de contraste específicas que llamamos *contraste objetivo*. Usando un fotómetro, medimos la intensidad de las zonas claras y oscuras de la imagen, esto es, el contraste de la imagen producida por la lente, la *medida de contraste de la imagen*. Repetimos las medidas para diferentes frecuencias espaciales, siempre con rejillas del mismo *contraste objetivo*.

Si realizamos la gráfica del resultado, con el eje horizontal como la frecuencia espacial y el eje vertical como la *medida de contraste de la imagen* como porcentaje del *contraste objetivo*, obtenemos la función de transferencia de la lente, esto es, cómo se transfiere el contraste a través de la lente. La Figura 2.18 muestra dos curvas correspondientes a la función de transferencia de dos lentes, una lente limpia (*clean lens*) y otra sucia (*buttered lens*).

Las imágenes naturales, a diferencia de las simples rejillas senoidales, están compuestas por multitud de frecuencias espaciales, ondas senoidales en diferentes orientaciones. Para encontrar las zonas que serán invisibles a la lente, podemos primero determinar la función de transferencia de esta, después analizar la imagen para extraer sus componen-

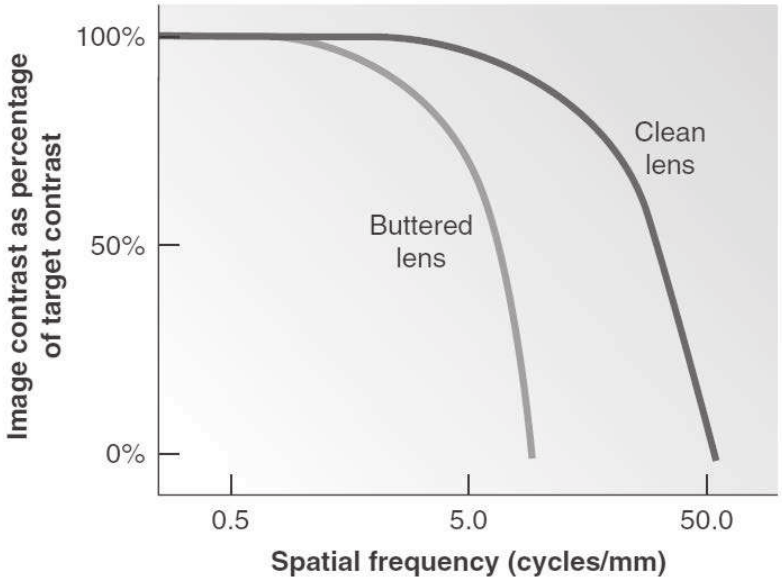


Figura 2.18: Dos funciones de transferencia para lentes. Cómo el contraste en la imagen formada por lentes se relaciona con el contraste en el objeto.

tes frecuenciales (mediante análisis de Fourier) y finalmente concluir qué componente de frecuencia espacial será visible para la lente y cuál no.

Ahora supongamos que en lugar de una lente tenemos el Sistema Visual Humano (HVS). La elección de qué frecuencias podemos percibir no es tan sencilla, puesto que determinar la función de transferencia de nuestro HVS es algo más complejo.

En el HVS no podemos reproducir el proceso empleado para obtener la función de transferencia tal y como se hace para las lentes, ya que la imagen se forma en el interior del ojo y eso es solo una pequeña parte de la función de transferencia del HVS, ya que además entran en juego procesos cognitivos y neurológicos.

Numerosos estudios se han encargado de caracterizar lo que llamamos la Función de Sensibilidad al Contraste (*Contrast Sensitivity Function*, CSF) a partir de mediciones del umbral de contraste en humanos para diferentes frecuencias espaciales [65]-[68], pero el más extendido y el que se ha usado en este trabajo corresponde al determinado por Daly [61].

La Figura 2.19 muestra la CSF para una persona adulta. El eje horizontal especifica la frecuencia espacial como el número de ciclos dentro de un grado de ángulo visual (ciclos por grado), mientras que el eje vertical representa el contraste mínimo requerido para ver

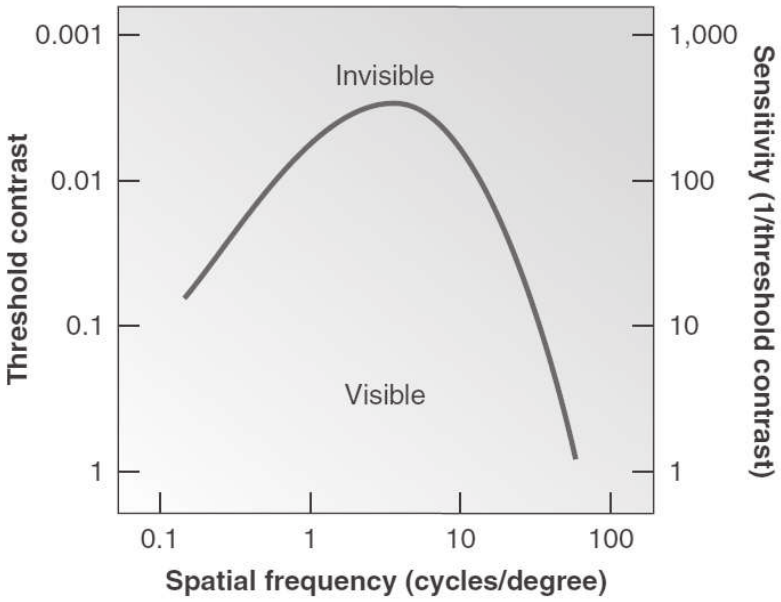


Figura 2.19: Curva de la Función de Sensibilidad al Contraste (CSF)

la rejilla. Esta curva define la ventana de visibilidad, siendo la parte superior de la curva trazada la combinación de contraste y frecuencia espacial que el ojo humano no puede percibir.

La curva CSF de la Figura 2.19 difiere de la función de transferencia de la lente de la Figura 2.18 a frecuencias bajas, ya que el HVS es menos sensitivo a frecuencias espaciales muy bajas que a frecuencias intermedias. Los objetos que se encuentren en una imagen cuya frecuencia espacial se aproxime a la óptima se visualizarán muy claramente, aun teniendo bajo contraste. En cambio, si el objeto tiene sus componentes frecuenciales muy bajas o elevadas, es posible que no se puedan visualizar aun teniendo buen nivel de contraste.

CSF en la codificación de vídeo

El enmascaramiento por contraste es una de las técnicas basadas en el HVS más utilizadas para reducir los artefactos de compresión. Consiste en incorporar la función de sensibilidad al contraste (CSF) durante la etapa de cuantificación en los códecs de imágenes y vídeos. Como ya hemos explicado, la CSF muestra que el HVS es incapaz de detectar diferencias entre objetos y su fondo bajo ciertas condiciones de luminancia, distancia o frecuencia espacial. Los artefactos de compresión pueden ser enmascarados bajo

estas condiciones, ya que funcionan como primer plano, mientras que la escena actúa como el fondo.

Al incorporar la CSF en el proceso de cuantificación, se pueden asignar diferentes niveles de cuantificación a las distintas componentes frecuenciales, reduciendo así los artefactos perceptibles y mejorando la eficiencia de compresión. Esto se debe a que las frecuencias espaciales a las que el HVS es menos sensible pueden ser cuantificadas de forma más agresiva sin afectar significativamente la calidad percibida de la imagen o el vídeo. De esta manera, se optimiza el uso del ancho de banda o espacio de almacenamiento al minimizar el impacto visual de la compresión.

A partir del estándar H.264 (AVC), y continuado en H.265 (HEVC), se incorpora en las especificaciones la posibilidad de incluir una matriz de pesos ponderada en la etapa de cuantificación. Por defecto, se utiliza una cuantificación uniforme; sin embargo, es posible seleccionar una matriz de cuantificación no uniforme basada en [61], así como permitir que el usuario defina su propia matriz de pesos (ver apartado 3.1.1).

2.3.2. Enmascaramiento (masking)

El enmascaramiento o *masking* es un fenómeno visual que ocurre cuando un estímulo que es visible por sí solo se vuelve invisible en presencia de otro estímulo [63].

Existe una relación entre ambos estímulos, la máscara o *masker* y el estímulo original. Algunas características similares en ambos estímulos causan la invisibilidad del estímulo original cuando se encuentra en presencia de la máscara; normalmente, esta interacción ocurre gradualmente a medida que las propiedades relacionadas cambian. Estas propiedades son la frecuencia espacial, la orientación y la fase de la máscara con respecto al estímulo original.

En algunas ocasiones ocurre el efecto opuesto, esto es, un estímulo hace visible otro estímulo que no era perceptible por sí solo.

Enmascaramiento por textura (texture masking)

El enmascaramiento espacial (también llamado enmascaramiento por textura o *texture masking*) es mayor cuando los estímulos tienen similares características (frecuencias

similares, orientaciones, colores, etc.) pero también ocurre entre estímulos de diferente orientación o de diferente frecuencia espacial.

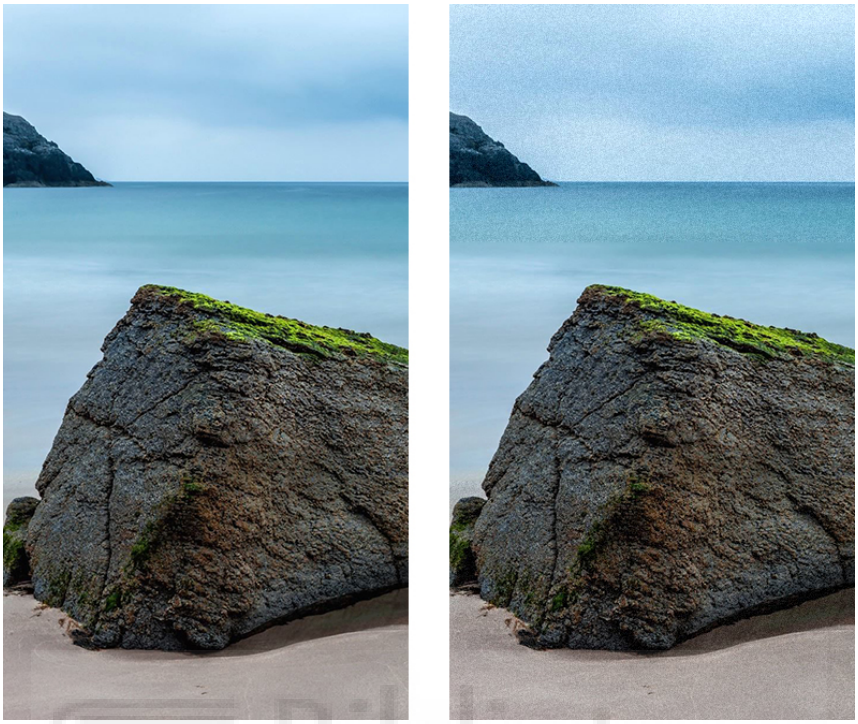


Figura 2.20: Ejemplo visual de enmascaramiento aplicando ruido a regiones con diferente textura. El fondo de la imagen actúa de máscara para el patrón de ruido añadido. A la izquierda se tiene la imagen original y a la derecha la imagen con un patrón de ruido añadido en la parte superior e inferior

Por ejemplo, en algunas regiones de una imagen algún ruido o artefactos debidos a la compresión son más visibles que en otras partes. En esos casos el fondo de la imagen actúa como máscara para estos artefactos. En la Figura 2.20 se muestran dos imágenes. A la izquierda se tiene la imagen original y a la derecha la misma imagen con ruido añadido en el tercio superior e inferior. Como se puede observar el ruido se hace visible en la parte superior de la imagen, sin embargo, a simple vista la parte inferior no muestra diferencias con respecto a la imagen original. Esto es debido a que la textura de la roca está enmascarando el ruido añadido.

En la Figura 2.21 se explica un ejemplo de los efectos del enmascaramiento por textura. La imagen muestra dos gráficas que representan la variación de luminancia a lo largo de una determinada orientación en la imagen. Si uno se sitúa al inicio del escalón, vemos cómo desde una luminancia inicial se eleva la misma en una distancia en píxeles deter-

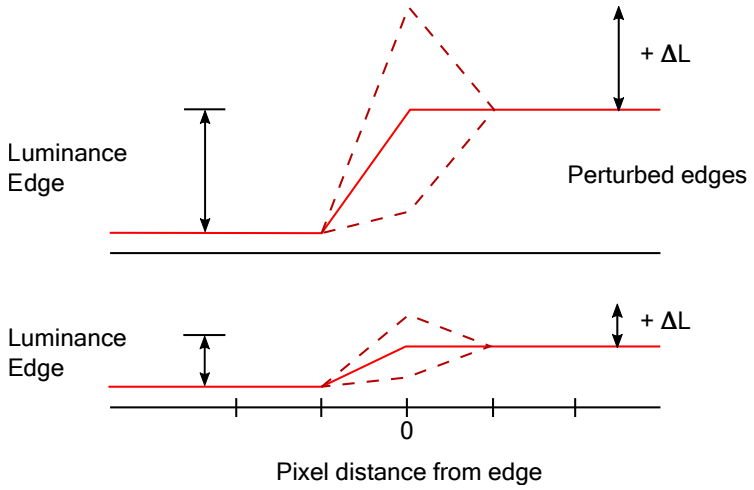


Figura 2.21: Efectos de enmascaramiento por textura (*spatial* o *texture masking*).

minada (eje x). Es decir, en esa orientación que representa la gráfica la luminancia crece hasta un determinado nivel en un determinado número de píxeles.

La gráfica superior indica una variación rápida en los niveles de luminancia en esa orientación. La gráfica inferior indica una variación lenta en los niveles de luminancia. Las líneas discontinuas representan la variación del umbral de visibilidad ΔL que la ley de Weber indica (Ecuación 2.7).

Por tanto, como vemos en la Figura 2.21, cuando hay un estímulo presente, en este caso una variación de luminancia grande, ΔL, en las cercanías de un punto en la imagen, la variación de luminancia en esa zona, por ejemplo, debido a una determinada de textura, hace que el umbral de detección suba y por tanto la sensibilidad para ver la distorsión disminuya. Así en zonas con alta textura (variación de luminancia en diversas orientaciones conjuntamente) tenemos menos posibilidad de ver las distorsiones presentes, porque la textura enmascara la distorsión. En cambio, cuando hay muy poca textura el umbral de detección baja con lo que la sensibilidad al contraste aumenta y por tanto detectaremos mejor la distorsión puesto que no hay textura que la enmascare.

El *texture masking* se ha estado utilizando en la codificación de imágenes con el objetivo de cuantificar en mayor medida las regiones donde el ojo humano es incapaz de detectar los artefactos producidos en el proceso codificación. Para ello, la gran mayoría de estudios se basan en clasificar primero los bloques de la imagen según su grado de textura (normalmente se clasifica en *TEXTURE*, *EDGE* y *PLAIN*) y después determinar un valor de sobre cuantificación a aplicar durante el proceso de transformación y cuanti-

ficación.

Aunque la mayoría de autores clasifican los bloques basándose en su información frecuencial, algunos autores, como [69] y [70], utilizan directamente la información espacial, es decir, el domino del píxel, para la clasificación, haciendo uso de algoritmos de detección de bordes como Canny, y determinando el grado de textura a partir de la proporción de *edge pixels* que contengan los bloques. Utilizando este método los autores son capaces de realizar una clasificación para cualquier tamaño de bloque de forma sencilla.

Más recientemente, algunos estudios utilizan redes neuronales convolucionales (CNN) para clasificar y/o sobre cuantificar los bloques directamente. En [71], Jin *et al.* proponen un modelo de JND (Just Noticeable Distortion) que adaptativamente reconoce y evalúa la calidad de la imagen, midiendo la distorsión causada por su JND. Además, utilizan un modelo de atención visual para segmentar la imagen y limitar la región donde se va a aplicar dicha distorsión.

Yuhao *et al.* [72] introducen un modelo JND basado en aprendizaje profundo no supervisado, utilizando redes neuronales convolucionales (CNN) que aprenden las características de redundancia visual del Sistema Visual Humano (HVS) sin datos pre-etiquetados. Para estimar el mapa JND del modelo propuesto, emplean la calidad de la imagen, un patrón de complejidad y la capacidad de enmascaramiento del ruido.

Por otro lado, los estudios basados en clasificar los bloques de imágenes en el dominio frecuencia toman como referencia el modelo descrito por Tong [26], que a su vez está basado en el estudio anterior realizado por Park [73]. Tong divide la imagen en bloques de 8×8 píxeles, debido a que su estudio se basa en añadir codificación perceptual al estándar JPEG. A cada bloque le realiza la transformada discreta del coseno (DCT-II) siguiendo las directrices del estándar JPEG, con lo que obtiene una matriz de coeficientes. Estos coeficientes se segmentan en tres grupos según su distancia a la componente continua, es decir, según su nivel frecuencial (Figura 2.22): LF (baja o *low frequency*), MF (media o *mid frequency*) y HF (alta o *high frequency*). Estos coeficientes se suman (o promedian, según el estudio), y tras un simple árbol de decisión, se determina el tipo de imagen al que pertenece el bloque, que puede ser *TEXTURE* (bloques que tienen muchas componentes frecuenciales complejas), *EDGE* (bloques que contienen bordes fuertemente marcados) o *PLAIN* (bloques generalmente *suaves*, con pocas componentes frecuenciales).

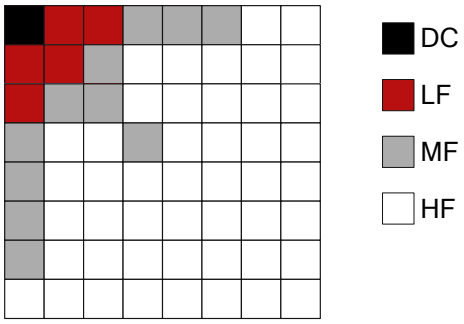


Figura 2.22: Clasificación de coeficientes transformados para bloques de 8×8. Diferenciamos LF (rojo), MF (gris) y HF (blanco) como baja, media y alta frecuencia respectivamente

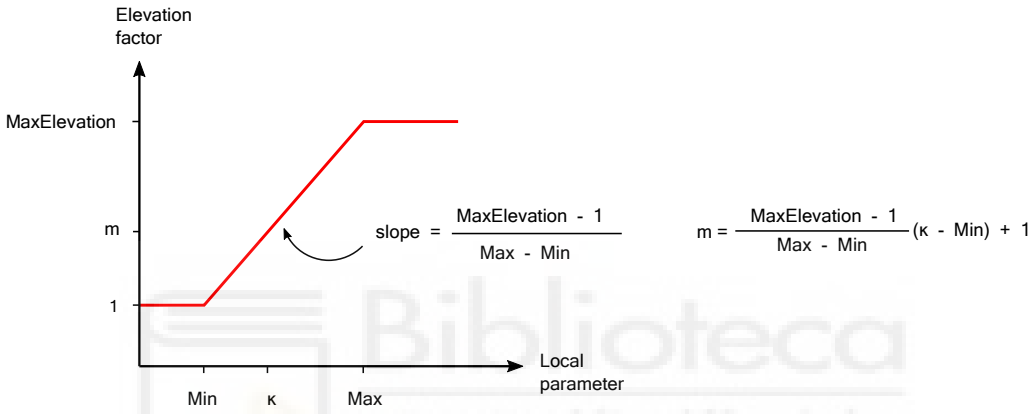


Figura 2.23: Gráfica del modelo lineal de elevación de umbral (*izquierda*) y expresión matemática (*derecha*)

Finalmente se obtiene un valor multiplicativo para el bloque k-ésimo, que elevará el nivel de cuantificación impuesto por el estándar. Según el estudio de Tong, para los bloques *TEXTURE* se obtiene el valor multiplicativo a partir de la siguiente Ecuación 2.8:

$$TextureMask(k) = (MaxTextureElevation - 1) \times \frac{TexEnergy(k) - MinEnergy}{MaxEnergy - MinEnergy} + 1 \quad (2.8)$$

donde $TexEnergy(k)$ corresponde a la suma de los valores MF y HF; el valor máximo posible del multiplicativo, $MaxTextureElevation$, se asigna a 2,25 y $MaxEnergy$ y $MinEnergy$ toman los valores 1800 y 290 respectivamente. Esta fórmula proviene de un simple modelo lineal de elevación de umbral (Figura 2.23).

Para los bloques clasificados como *EDGE* no se utiliza la fórmula anterior, sino que se aplica un nivel de elevación de *masking* más restrictivo. Si $LF + MF <= 400$, se determina

un valor 1,125, en caso contrario, se aplica un valor de 1,25.

Los bloques clasificados como *PLAIN* no aplican ningún valor de *texture masking*, ya que, en estos bloques, al no tener textura, cualquier artefacto debido a una mayor cuantificación será perceptible por el ojo humano.

Zhang *et al.* [27][74] publican una versión mejorada del clasificador de bloques de Tong. En su algoritmo, aparte de utilizar la suma absoluta de los coeficientes de la transformada agrupados según sean LF, MF y HF, utiliza también su valor promedio a lo largo del algoritmo matemático para determinar la energía de textura y así poder clasificar de forma más eficiente los bloques de 8×8. El algoritmo de Zhang clasifica la imagen evaluando la energía de textura y la presencia de bordes fuertemente marcados.

Si un bloque tiene mucha o poca energía de textura se clasifica como *TEXTURE* o *PLAIN* respectivamente. En cambio, si tiene valores intermedios se tendrá también en cuenta la presencia de bordes para determinar si el borde es clasificado como *TEXTURE*, *EDGE* o *PLAIN*.

Una vez clasificado el bloque, Zhang obtiene el factor de enmascaramiento a_e a partir de dos partes o efectos: *inter-band* e *intra-band masking*. El efecto de *Inter-band masking* ξ se estima considerando la fuerza de la máscara así como la diferencia (de valor y orientación) entre la máscara y lo enmascarado, y este autor lo calcula siguiendo la metodología de Tong [26]: para los bloques *TEXTURE* utiliza un modelo de elevación de umbral (Ecuación 2.8); para los bloques *EDGE* dependiendo de los coeficientes toma unos valores fijos más restrictivo; y si es un bloque *PLAIN* no aplica nada para no empeorar la calidad de la imagen.

El efecto *Intra-Band masking* se refiere a la tolerancia al error imperceptible ocasionada por la señal en su misma sub-banda. Es decir, para cada coeficiente de la transformada (i, j) dentro del bloque (n_1, n_2) se ha estudiado la forma de aumentar el nivel de cuantificación en función de su posición y del valor del propio coeficiente, de forma que este aumento sea imperceptible por el sistema visual humano. El factor multiplicativo para el enmascaramiento por textura queda finalmente determinado como:

$$a_e(n_1, n_2, i, j) = \begin{cases} \xi(n_1, n_2) & \text{for } (i, j) \in LF \cup MF \\ & \text{in EDGE block,} \\ \xi(n_1, n_2) & \text{otherwise,} \\ \times \max\left\{1, \left(\frac{C(n_1,n_2,i,j)}{t_b(n_1,n_2,i,j)}\right)^{\varepsilon}\right\} & \end{cases} \tag{2.9}$$

donde $C(n_1, n_2, i, j)$ son los coeficientes de la DCT y $\varepsilon = 0{,}36$. Dado que el HVS es más sensible a los cambios en los bordes, el efecto *Intra-band masking* no se aplica a los coeficientes de baja y media frecuencia para los bloques *EDGE* para evitar la sobreestimación.

3. ANÁLISIS PERCEPTUAL DE LAS DIFERENTES HERRAMIENTAS DE CODIFICACIÓN DE HEVC

El codificador de vídeo High Efficiency Video Coding (HEVC) es uno de los estándares más recientes desarrollados conjuntamente por los organismos de normalización ITU-T e ISO/IEC en la llamada Joint Collaborative Team on Video Coding (JCT-VC) [54]. Desde su primera versión, en 2013, hasta la última del 2021, se han ido publicando numerosas revisiones de la norma, incluyendo mejoras en las diferentes etapas de codificación, así como nuevos perfiles y extensiones, mejorando su capacidad y rendimiento en aplicaciones concretas.

El objetivo principal del HEVC es el de reducir la tasa binaria hasta un 50 % manteniendo la calidad perceptual, en comparación con el estándar previo H.264/AVC, sin incrementar la complejidad del codificador. Para ello, se introdujeron técnicas avanzadas de codificación, algunas heredadas y otras innovadoras, como el particionado Quad-tree y el filtro Sample Adaptive Offset (SAO).

De entre las numerosas herramientas de codificación incluidas en el HEVC, algunas de ellas tienen en consideración el comportamiento no lineal del sistema visual humano (HVS), buscando optimizar la calidad subjetiva durante el proceso de codificación. Un ejemplo de ello, desarrollado más adelante, es el Scaling List, que aplica una cuantificación no uniforme sobre los coeficientes transformados. Aunque estas técnicas han logrado reducir efectivamente la tasa de bits, no está garantizado que estén optimizadas perceptualmente, ya que se desarrollaron utilizando métricas totalmente objetivas, como la PSNR, que no reflejan de manera precisa la evaluación perceptual de la calidad [3], [4], [8], [9].

Para evaluar adecuadamente el rendimiento perceptual de las herramientas de codificación del estándar HEVC, es crucial emplear métricas que estén altamente relacionadas con la calidad que percibimos los seres humanos. Es por ello por lo que se ha realizado un estudio para analizar el rendimiento perceptual de las diferentes herramientas de codificación incluidas en el software de referencia HEVC Test Model (HM) [75], codificando un conjunto normativo de secuencias de vídeo con diferentes configuraciones para entender

cuál maximiza, en promedio, la calidad perceptual. Este análisis no solo considera la contribución individual de cada herramienta de codificación sino también su rendimiento en conjunto con otras, utilizando métricas objetivas basadas en modelos de percepción, así como la métrica de calidad Video Multimethod Assessment Fusion (VMAF) desarrollada por Netflix [76].

La contribución principal de este trabajo reside en el análisis de rendimiento de varias herramientas de codificación HEVC en términos de su impacto en la calidad perceptual del vídeo decodificado. Este análisis exhaustivo revela resultados que difieren de los proporcionados por la PSNR, lo cual será crucial para futuros estudios y configuraciones de codificadores que busquen maximizar el rendimiento perceptual.

3.1. Herramientas de codificación del HEVC

El estándar HEVC incorpora numerosas herramientas de codificación que pueden ser habilitadas, modificadas o deshabilitadas, editando determinados parámetros en el archivo de configuración o por línea de comandos, con el objetivo de optimizar la calidad de reconstrucción, reducir el tamaño del flujo de bits o simplificar la complejidad del codificador.

Estos parámetros permiten ajustar distintos aspectos del proceso de codificación, como la estructura de codificación, la estimación de movimiento, la cuantificación, el codificador entrópico, filtros, el rate-control, entre otros [77]. En nuestro estudio, se han seleccionado las herramientas de codificación que tienen un mayor impacto en la calidad visual del vídeo reconstruido, como son el Scaling List, In-loop filters (Deblocking y SAO), Rate-Distortion Optimization Quantization, Transform Skip y Sign Data Hidding.

3.1.1. Scaling List

El Sistema Visual Humano (HVS) no percibe todas las frecuencias espaciales por igual [63], siendo menos sensible a frecuencias muy bajas y muy elevadas. El parámetro ScalingList en el estándar HEVC implementa la llamada función de sensibilidad al contraste (CSF), que modifica la matriz de cuantificación al variar los pesos de los coeficientes según la posición en la que se encuentren.

El parámetro ScalingList tiene tres modos de funcionamiento. El primer modo (ScalingList = 0) deshabilita el uso del escalado basado en la CSF, siendo las matrices de cuantificación uniformes (ver Figura 3.1-a).

En el segundo modo (ScalingList = 1), se utilizan matrices de cuantificación dependientes de la frecuencia, que se ajustan mejor a la subjetividad del sistema visual humano al permitir cuantificar más intensamente los coeficientes asociados a frecuencias más altas (ver Figura 3.1-b). El estándar utiliza los resultados del estudio de Daly [61] para determinar los pesos de la matriz de cuantificación.

$$\begin{bmatrix} 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \end{bmatrix} \qquad \begin{bmatrix} 16 & 16 & 16 & 16 & 17 & 18 & 21 & 24 \\ 16 & 16 & 16 & 16 & 17 & 19 & 22 & 25 \\ 16 & 16 & 17 & 18 & 20 & 22 & 25 & 29 \\ 16 & 16 & 18 & 21 & 24 & 27 & 31 & 36 \\ 17 & 17 & 20 & 24 & 30 & 35 & 41 & 47 \\ 18 & 19 & 22 & 27 & 35 & 44 & 54 & 65 \\ 21 & 22 & 25 & 31 & 41 & 54 & 70 & 88 \\ 24 & 25 & 29 & 36 & 47 & 65 & 88 & 115 \end{bmatrix}$$

(a) matriz de cuantificación constante (b) matriz de cuantificación escalada

Figura 3.1: Matrices 8×8 de cuantificación intra-frame definidas en HM para (a) ScalingList = 0 y (b) ScalingList = 1.

Existe un tercer modo de funcionamiento (ScalingList = 2), en el cual se permite la incorporación de matrices de cuantificación personalizadas a partir de la lectura de un fichero de configuración propio.

El estándar HEVC define las matrices de cuantificación no uniformes para tamaños de bloque 4×4 y 8×8, tanto para las componentes de luminancia y crominancia como para el tipo de codificación, ya sea intra-frame o inter-frame. A la hora de aplicar estas matrices en bloques de tamaño superior (16×16 y 32×32), se realiza un sobre muestreo de la matriz 8×8, replicando los valores tal y como se observa en la Figura 2.15.

3.1.2. Filtro de Deblocking

El filtro Deblocking en HEVC se aplica en los píxeles limítrofes a las unidades de predicción (PU) y de transformación (TU). Este filtro viene heredado del estándar H.264/AVC,

aunque para este codificador se ha mejorado y simplificado su algoritmo. El objetivo del filtro de Deblocking es el de reducir los artefactos causados por la naturaleza de la codificación basada en bloques, como puede ser el efecto tablero de ajedrez.

Este filtro se aplica de manera adaptativa en diferentes grados (normal o fuerte), o puede no aplicarse, dependiendo de la relación entre los dos bloques adyacentes, y una serie de condicionantes que se evalúan en cada caso, como el tipo de predicción, si existen coeficientes de transformación distintos de cero, índices de referencia o vectores de movimiento distintos [78].

Dependiendo del resultado de evaluar dichos condicionantes y del parámetro de cuantificación promedio, se determinan una serie de umbrales a partir de una tabla predefinida, y se decide aplicar uno de los tres grados de filtrado para el canal de luminancia, mientras que, para crominancia, se decide entre no filtrar y un filtrado normal.

En la Figura 3.2 se observa como la aplicación del filtro de deblocking reduce el efecto de la compresión de bloques, mejorando la calidad perceptual de la imagen codificada.

Figura 3.2: Ejemplo de aplicación del filtro Deblocking para una región aumentada de un frame de la secuencia BlowingBubbles, codificado con QP=37: (a) DB deshabilitado, (b) DB habilitado.

En [78], los autores afirman que el uso del filtro de deblocking mejora tanto la calidad subjetiva como la objetiva de las secuencias de vídeo decodificadas.

3.1.3. Filtro SAO

El filtro SAO (Sample Adaptive Offset) es un proceso incorporado en el estándar de codificación HEVC que se dedica a modificar las muestras decodificadas para mejorar la calidad de las imágenes reconstruidas, particularmente en lo que respecta a la nitidez de los bordes y la reducción de artefactos de bandeado (banding artifacts) y anillo (ringing artifacts) [79]. Este filtro se aplica después del filtro de Deblocking, trabajando en regiones específicas de la imagen, las cuales se alinean con las CTUs.

Las operaciones del filtro SAO son de carácter no lineal y se aplican condicionalmente, añadiendo un valor de desplazamiento (offset) a cada muestra decodificada basado en tablas de búsqueda transmitidas por el codificador [62]. Estos desplazamientos pueden ser tanto positivos como negativos y están determinados por el codificador, generalmente bajo criterios que optimizan el rate-distortion. Cada CTU lleva sus propios parámetros SAO y estos pueden marcarse para ser heredados de CTUs adyacentes, lo cual optimiza la señalización.

El filtro SAO opera en dos modos principales: band offset (BO) y edge offset (EO). En el modo BO, la amplitud completa de la muestra se divide uniformemente en 32 bandas. Las muestras que pertenecen a cuatro de estas bandas consecutivas se modifican mediante la adición de valores de desplazamiento transmitidos. Estos desplazamientos están limitados en un rango de -7 a 7 (para 8 bits de profundidad), y el signo de cada desplazamiento se envía por separado en el bitstream.

Por otro lado, el modo EO se utiliza para la clasificación de desplazamiento de bordes y opera con base en gradientes horizontales, verticales o diagonales dentro de la CTU. Cada muestra dentro de una CTU es clasificada mediante la comparación de valores de muestras decodificadas en una de las siguientes cinco categorías: mínimo local, borde positivo, área plana, borde negativo y máximo local. Los valores de desplazamiento asociados a estas categorías son siempre positivos para las categorías de mínimo local y borde negativo, y negativos para las categorías de máximo local y borde positivo, lo que resulta en un efecto general de suavizado.

En la Figura 3.3 se observa como el uso del filtro SAO reduce notablemente los efectos producidos por la codificación, suavizando las regiones con bordes definidos, lo cual

Figura 3.3: Ejemplo de aplicación del filtro SAO para una región aumentada de un frame de la secuencia RaceHorses, codificado con QP=32: (izquierda) SAO habilitado, (centro) SAO deshabilitado, (derecha) Imagen original [62].

proporciona una imagen perceptualmente más clara.

En [62], los autores afirman que el uso del filtro SAO puede proporcionar unas ganancias promedio de codificación del 3,5 %. Para medir esta ganancia, han utilizado la métrica Bjøntegaard-Delta Rate (BD-Rate) [7], que utiliza la PSNR, una métrica no subjetiva. En cuanto a la calidad subjetiva, los autores afirman que, basándose en experimentos realizados internamente, en general se percibe una mejora de la calidad.

3.1.4. Rate-Distortion Optimized Quantization (RDOQ)

El algoritmo Rate-Distortion Optimized Quantization (RDOQ) en HEVC busca identificar el conjunto óptimo de coeficientes cuantificados transformados que representan los datos residuales de un bloque codificado. Este proceso se realiza minimizando una función Lagrangiana (Ecuación 3.1) y tomando en cuenta tanto el error de cuantificación (distorsión, D) como la cantidad de bits necesarios para transmitir los coeficientes (B).

$$RD = D + \lambda \cdot B \tag{3.1}$$

El algoritmo RDOQ se divide en tres etapas: cuantificación de coeficientes transformados, eliminación de grupos de coeficientes (CG) y selección del último coeficiente distinto de cero [80].

En la primera etapa, por cada coeficiente transformado se calcula un valor denominado Level. Luego, considera dos magnitudes adicionales del coeficiente cuantificado: Level-1 y 0. Para cada magnitud, se calcula su coste RD y se selecciona la magnitud que menor

valor de RD obtenga.

En la segunda etapa, cada TU es dividido en grupos de coeficientes (coefficient groups, CG) de tamaño 4×4. Para cada CG, el codificador calcula el coste RD de eliminar completamente dicho CG, es decir, marcar todos los coeficientes de ese CG a cero. Si la eliminación resulta en una reducción de coste, el CG seleccionado se elimina. Este proceso puede resultar en una reducción significativa de la tasa de bits, aunque introduce distorsión en la imagen reconstruida.

Tras los dos pasos anteriores, para los CGs restantes en cada TU, se analiza los coeficientes para encontrar la mejor posición del último coeficiente distinto de cero en términos de coste RD.

Durante la operación de RDOQ, el codificador calcula el coste de cada conjunto considerado de coeficientes o grupos de coeficientes. Este coste (RD) se calcula considerando el número de bits requeridos para codificar el coeficiente, los CGs o la TU seleccionada, la distorsión introducida y ambos valores son ponderados por un multiplicador Lagrangiano (λ).

Dado que en el estándar HEVC solamente la sintaxis y semántica del bitstream están estandarizados, no existe una definición única del algoritmo RDOQ. En la implementación del algoritmo RDQO utilizada en el modelo de test del HEVC (HM) [75], el codificador utiliza valores estimados para el cálculo de la distorsión y del número de bits necesarios, lo que acelera las operaciones del codificador aunque con algunos errores en la selección de coeficientes y una ligera degradación del rendimiento de compresión.

Siendo el algoritmo RDOQ un método eficaz en términos de aumentar el rendimiento rate-distortion, en [25], los autores afirman que la mejora de la calidad de reconstrucción lograda por esta técnica, basada en la métrica PSNR para el cálculo de la distorsión, es perceptualmente insignificante en cuanto a cómo el observador interpreta la calidad percibida de los datos de vídeo comprimidos.

3.1.5. Transform Skip

El parámetro Transform Skip del estándar HEVC permite que el codificador omita la etapa de transformación, de manera que los errores de predicción se codifiquen directa-

mente en el dominio espacial [81]. Esta funcionalidad resulta particularmente beneficiosa para la compresión de secuencias de vídeo sintéticas, como las utilizadas en escritorio remoto o presentaciones de diapositivas, que contienen predominantemente texto y gráficos [82].

El modo Transform Skip está restringido a TUs de tamaño 4×4 en el modo intra-frame. Este modo es especialmente útil para regiones o bloques con muchos bordes marcados, como los encontrados en la codificación de contenido sintético o artificial (clase F), logrando ganancias significativas. Excepto por la adición de una señalización que indique si un TU Intra 4×4 utiliza Transform Skip, no hay cambios significativos en otros aspectos del proceso de codificación.

En términos de calidad visual, los vídeos reconstruidos mediante el modo Transform Skip presentan bordes más nítidos, menos artefactos y mayores detalles en comparación con los vídeos reconstruidos sin este modo, tal y como se observa en la Figura 3.4.

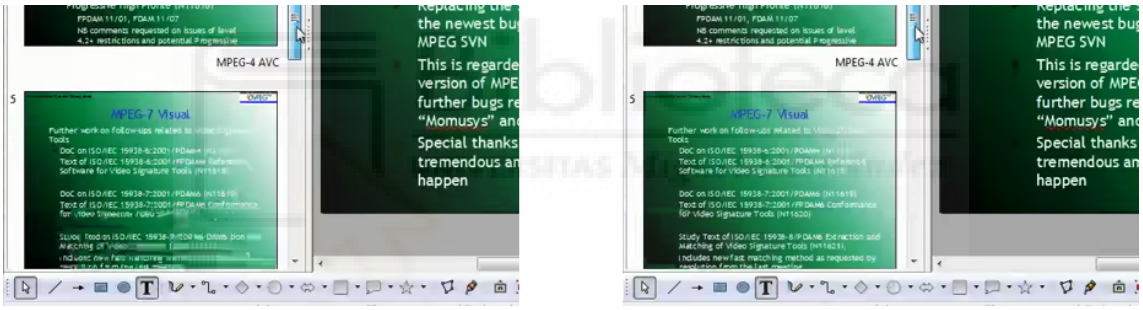


Figura 3.4: Ejemplo de aplicación del modo Transform Skip para una región aumentada de un frame de la secuencia SliceEditing, codificado con QP=37: (izquierda) TS deshabilitado, (derecha) TS habilitado [82].

Cabe mencionar que, aunque este método puede aumentar el tiempo de codificación en aproximadamente un 30 %, el ahorro de bits sigue siendo significativo, y para el decodificador, el impacto en tiempos de ejecución es mínimo.

Además, el software de referencia HM incluye un parámetro adicional llamado TransformSkipFast. Este parámetro permite habilitar una reducción de pruebas en la toma de decisión del modo Transform Skip, acelerando así el proceso de codificación.

3.1.6. Sign data hidding

El algoritmo Sign Data Hiding (SDH) o Sign Bit Hiding (SBH) en el estándar HEVC proporciona una técnica alternativa para codificar el signo del primer coeficiente distinto de cero de un bloque residual [83]. Este algoritmo no siempre escribe explícitamente el signo en el bitstream, sino que se infiere a partir de la paridad de la suma de los coeficientes cuantificados, según una convención predefinida.

Durante el proceso de codificación, si la paridad no coincide con el signo del primer coeficiente distinto de cero, el codificador ajusta uno de los coeficientes cuantificados para obtener la paridad deseada. Este ajuste se realiza modificando el coeficiente que resulte en la menor penalización de rate-distortion (R/D).

14	3	7	**1**
9	-4	1	0
2	-4	0	0
-1	0	1	0

(a)

14	3	7	**0**
9	-4	1	0
2	-4	0	0
-1	0	1	0

(b)

Figura 3.5: Ejemplo de cómo el algoritmo Sign Data Hiding modifica el valor de un coeficiente para un TU de tamaño 4×4 para ajustar la paridad del signo del primer coeficiente distinto de cero.

En la Figura 3.5 se muestra un ejemplo práctico de aplicación de este método. En la Figura 3.5(a) se muestra un sub-bloque transformado de tamaño 4×4, cuya suma absoluta de los coeficientes es 47. Por convenio, el valor impar de esta suma derivaría en un signo negativo para el primer coeficiente distinto a cero. Si este no es el caso, el codificador cambia el valor de un coeficiente (en negrita) de manera que la suma absoluta sea par (Figura 3.5(b)). La selección específica del coeficiente a modificar se realiza mediante criterios de rate-distortion, eligiendo el coeficiente que tenga el menor coste R/D.

En cuanto al rendimiento, la técnica de SDH logra una reducción promedio de BD-Rate del 0.6 % para el modo de codificación All Intra. Cabe mencionar que, aunque modificar los valores de los coeficientes puede aumentar la distorsión, la ganancia en BD-Rate se obtiene principalmente gracias a la reducción de tasa proporcionada por esta técnica y no debido a un aumento en la calidad.

Tabla 3.1: Secuencias de testeo para HEVC.

Clase	Nombre de secuencia	Resolución	Número de frames	Tasa de frames	Profundidad de bits
A	Traffic	2560x1600	150	30	8
	PeopleOnStreet		150	30	8
	Nebuta		300	60	10
	SteamLocomotive		300	60	10
B	Kimono	1920x1080	240	24	8
	ParkScene		240	24	8
	Cactus		500	50	8
	BQTerrace		600	60	8
	BasketballDrive		500	50	8
C	RaceHorses	832x480	300	30	8
	BQMall		600	60	8
	PartyScene		500	50	8
	BasketballDrill		500	50	8
D	RaceHorses	416x240	300	30	8
	BQSquare		600	60	8
	BlowingBubbles		500	50	8
	BasketballPass		500	50	8
E	FourPeople	1280x720	600	60	8
	Johnny		600	60	8
	KristenAndSara		600	60	8
F	BaskeballDrillText	832x480	500	50	8
	ChinaSpeed	1024x768	500	30	8
	SlideEditing	1280x720	300	30	8
	SlideShow	1280x720	500	20	8

3.2. Métodos y procedimientos

En este estudio se siguen las indicaciones establecidas por la Common Test Condition [84], que proporciona un marco regulatorio con secuencias definidas y varias configuraciones base para el software de referencia HM. Las secuencias de testeo se dividen en seis grandes grupos (A–F), representando las clases de la A a la D diferentes contenidos, resoluciones, velocidades de fotogramas y profundidades de bits. La clase E se centra en vídeos basados en videoconferencias, mientras que la clase F se dedica a vídeos sintéticos, contenido generado por ordenador y contenido de aplicaciones de escritorio. El conjunto de secuencias de testeo utilizadas se definen en la Tabla 3.1.

Este trabajo se centra exclusivamente en el perfil de codificación All Intra Main (AI Main), por lo que no se ha realizado procesamiento ni análisis temporal. Se ha utilizado únicamente este perfil de codificación por los siguientes motivos. Por un lado, la mayoría de las métricas perceptuales están disponibles únicamente para imágenes, es decir, analizan cada fotograma de las secuencias de vídeo de forma independiente, obteniendo finalmente un valor de calidad promedio. Por otro lado, las herramientas de codificación analizadas influyen en la calidad de la imagen reconstruida debido a un proceso de predicción. En dicho proceso, se cuantifica un error residual obtenido a partir de una predicción (espacial o temporal). Este error cuantificado afecta a la calidad final, independientemente del tipo de predicción utilizada. Es decir, la calidad final depende de la precisión de la predicción, no del tipo de predicción realizada.

Para analizar el rendimiento R/D de las herramientas de codificación, se ha utilizado la métrica BD-Rate, definida por Bjøntegaard [7], ampliamente utilizada a la hora de comparar curvas R/D. Se siguieron las instrucciones del HEVC conformance test standard [84], usando los valores de calidad QPs 22, 27, 32 y 37 para conformar las curvas R/D. También se añadió un QP adicional (QP = 42) para adaptarse mejor a la respuesta de rango dinámico de otras métricas perceptuales objetivas, proporcionando resultados BD-Rate más precisos.

Para evaluar la influencia de cada parámetro en la calidad perceptual, todas las secuencias de testeo se codificaron activando y desactivando todas las herramientas de codificación analizadas, obtenido un total de 64 posibles combinaciones de configuración. Estas configuraciones se ejecutaron con la versión 16.20 del software de referencia del HEVC (HM) [75].

Dada la gran cantidad de mediciones a realizar, se descartó el uso de pruebas subjetivas como DMOS, optando por obtener valoraciones objetivas basadas en métricas de calidad perceptuales ampliamente utilizadas por la comunidad científica, como son la SSIM, MS-SSIM, VIF, PSNR-HVS-M y VMAF.

La SSIM (Structural Similarity) [3], MS-SSIM (Multi-Scale Structural Similarity) [4], VIF (Visual Information Fidelity) [85] y PSNR-HVS-M [86] son métricas que intentan caracterizar la subjetividad del Sistema Visual Humano (HVS) sin incluir información temporal en sus algoritmos de evaluación de calidad.

La métrica VMAF, desarrollada por Netflix [76], utiliza técnicas de aprendizaje automático para estimar los resultados de pruebas subjetivas. Ha mostrado una alta correlación con la calidad perceptual en diversos estudios [87]-[89]. A la hora de aplicar esta métrica en nuestro estudio, hemos optado por deshabilitar la componente de análisis temporal, y así mantener la coherencia con la configuración experimental y evitar efectos indeseados al comparar los resultados con las otras métricas de calidad perceptuales.

Con respecto a las curvas R/D obtenidas, las curvas de referencia se han obtenido utilizando la configuración por defecto del perfil de codificación AI Main, cuyos valores se muestran en la Tabla 3.2.

Tabla 3.2: Configuración por defecto en el perfil All Intra Main.

Parámetro	**Valor**
QP	22, 27, 32, 37, 42
ScalingList	0
LoopFilterDisable	0
SAO	1
RDOQ	1
RDOQTS	1
TransformSkip	1
TransformSkipFast	1
SignHideFlag	1

3.3. Resultados

En esta sección se van a mostrar los resultados obtenidos tras codificar el conjunto de secuencias de testeo, habilitando y deshabilitando las diferentes herramientas de codificación descritas anteriormente, en comparación con la configuración predeterminada de perfil AI Main.

Para medir el rendimiento R-D de las diferentes configuraciones, hemos utilizado la métrica BD-Rate. Los resultados de las Tablas 3.3 a 3.8 son proporcionados por dicho método al utilizar como métrica base las diferentes métricas perceptuales descritas anteriormente. Los valores negativos en estas tablas corresponden a reducciones de BD-Rate, es decir, a ganancias perceptuales, y los valores positivos corresponden a incrementos de

BD-Rate o pérdidas perceptuales, con respecto a la configuración predeterminada.

En el artículo [10], anexo a esta tesis, se muestran las tablas completas de resultados para cada clase y para cada combinación de estado de las herramientas de codificación. En esta sección, sin embargo, se ha elaborado un resumen de dichas tablas para facilitar su lectura y análisis posterior. Estas tablas resumen se han creado para cada herramienta de codificación analizada, promediando las diferencias de deshabilitar/habilitar dicha herramienta de codificación para cada una de las combinaciones posibles.

Para facilitar aún más la lectura de los datos, se han resaltado las celdas utilizando un mapa de calor. Cuanto mayor sea la ganancia, más verde será la celda, y cuanto mayor sea la pérdida, más amarilla será la celda.

Además, se ha incluido un análisis de tiempos de codificación para cada una de las coding tools tratadas, con el objetivo de estimar su coste computacional. Considerando ambas métricas, la del rendimiento perceptual y la del coste computacional, se puede proponer una solución óptima que ofrezca mejores resultados perceptuales con rendimiento computacional equilibrado.

En las siguientes subsecciones, describiremos los resultados obtenidos, mostrando el comportamiento perceptual de cada herramienta de codificación bajo estudio, y en la siguiente sección, se proporcionará un análisis y discusión de estos resultados.

3.3.1. Scaling List

Al habilitar el parámetro ScalingList se utiliza una cuantificación no uniforme basada en la función de sensibilidad al contraste (CSF). Por defecto, esta función está deshabilitada, por lo que hemos analizado su influencia perceptual al habilitarla. En la Tabla 3.3 se muestra, en promedio, dicho impacto perceptual en el rendimiento de codificación.

Se puede observar que la clase B destaca al tener los valores más bajos en todas las métricas, lo que indica que las secuencias de alta resolución (1920x1080) experimentan una mejora notable en la calidad perceptual al habilitar este parámetro. Con un valor de $-1{,}89\,\%$ para la métrica VIF y $-2{,}65\,\%$ para la métrica PSNR-HVS-M, esta clase se beneficia significativamente de la activación del parámetro ScalingList.

En contraste, la clase D presenta un valor positivo para la métrica SSIM ($0{,}91\,\%$),

Tabla 3.3: Rendimiento perceptual promedio al habilitar ScalingList [% BD-Rate].

Class	SSIM	MS-SSIM	VMAF	VIF	PSNR HVS-M
A	−0,84	−0,54	−0,73	−0,88	−1,32
B	−1,59	−1,16	−1,66	−1,89	−2,65
C	−0,71	−1,02	−1,58	−0,91	−1,25
D	0,91	−0,61	−1,45	−0,79	−1,11
E	−1,03	−0,78	−0,86	−0,84	−1,35
F	−0,68	−0,4	−0,75	−0,52	−0,71

aunque las demás métricas siguen siendo negativas. Esto sugiere que, aunque puede haber una leve disminución en la calidad perceptual según la SSIM para las secuencias de baja resolución, las otras métricas aún indican mejoras perceptuales.

Por último, la clase F, relacionada con contenidos generados por ordenador y capturas de pantalla, presenta los valores más bajos entre todas las clases, aunque siguen siendo positivos. Esto podría indicar que, aunque hay una mejora al habilitar ScalingList, no es tan pronunciada como en las otras clases.

Como conclusión, habilitar el parámetro ScalingList parece ofrecer mejoras en calidad perceptual para casi todas las clases, siendo especialmente notables en las secuencias de alta resolución. En cuanto a su complejidad de codificación, al activar la Lista de Escalado, el tiempo medio de codificación aumenta entre un 3,47 % y un 8,44 %, según la cuantificación aplicada, como se muestra en la Tabla 3.9.

3.3.2. Filtro de Deblocking

El filtro de Deblocking atenúa el efecto de bloques causado por la partición de las imágenes durante el proceso de codificación. Al deshabilitar este filtro, los artefactos de bloques se vuelven visibles a medida que se aumenta el valor de la QP.

La tabla 3.4 refleja el impacto promedio en el rendimiento perceptual cuando se deshabilita el filtro de Deblocking.

Un aspecto clave que salta a la vista es el impacto especialmente negativo en las secuencias de clase E, con valores considerablemente altos en métricas como la MS-SSIM y la PSNR-HVS-M, llegando a 2,76 % y 3,49 % respectivamente. Esto sugiere que

Tabla 3.4: Rendimiento perceptual promedio al deshabilitar el Filtro de Deblocking [% BD-Rate].

Class	SSIM	MS-SSIM	VMAF	VIF	PSNR HVS-M
A	0,23	1,22	0,12	−0,43	2,15
B	0,24	1,67	0,14	−0,08	2,88
C	0,77	1,25	−0,02	0,4	2,36
D	2,33	0,51	−0,09	0,07	1,4
E	2,57	2,76	0,47	0,4	3,49
F	1,24	1,59	−0,03	0,25	2

las secuencias de alta resolución destinadas a contenidos televisivos (como entrevistas o informativos) podrían ser las más afectadas perceptualmente al deshabilitar el filtro de Deblocking.

Las secuencias de clase D, que corresponden a una resolución más baja, parecen ser menos sensibles a la omisión de este filtro en comparación con otras clases, especialmente si observamos métricas como MS-SSIM y VMAF. A pesar de esto, los valores no son los más bajos del conjunto. Es la métrica VIF, y solo para las secuencias de clase A, donde se obtiene una ganancia perceptual promedio de un −0,43 %.

En cuanto a la clase F, que incluye contenidos generados sintéticos y artificiales, aunque presenta una variación notable en algunas métricas, no parece ser la más afectada en general al deshabilitar el filtro, especialmente en comparación con la clase E.

Resulta evidente que deshabilitar el filtro de Deblocking tiene como consecuencia una pérdida de calidad perceptual generalizada para todas las clases y secuencias de vídeo, sobre todo para las secuencias de clase E. En cuanto a la complejidad de codificación, al desactivar el filtro DB se observa una reducción promedio del 0,3 % en el tiempo de codificación, según la Tabla 3.9.

3.3.3. Filtro SAO

El filtro SAO busca minimizar la distorsión introducida principalmente por la etapa de cuantificación. La tabla 3.5 nos proporciona un vistazo claro sobre cómo se ve afectado el rendimiento perceptual cuando se deshabilita este filtro.

Tabla 3.5: Rendimiento perceptual promedio al deshabilitar el Filtro SAO [% BD-Rate].

Class	SSIM	MS-SSIM	VMAF	VIF	PSNR HVS-M
A	0,11	0,36	−3,14	−0,1	0,83
B	0,22	0,67	−3,12	−0,23	1,16
C	0,66	0,83	−2,87	0,28	1,38
D	0,98	0,22	−2,64	0,14	0,8
E	0,9	1,18	−3,14	0,27	1,41
F	1,45	1,89	0,19	2,05	1,97

La decisión de deshabilitar el filtro SAO en el estándar HEVC resulta en una degradación de la calidad perceptual para todas las métricas a excepción de la métrica VMAF. Para esta métrica, las ganancias que se obtienen al no utilizar el filtro SAO en secuencias naturales son considerables, sobre todo para las secuencias de vídeo de alta resolución, donde se alcanza una ganancia promedio del −3,14 %.

Por otro lado, se puede destacar que las secuencias de clase F, que comprenden contenidos generados por ordenador, presentan los valores más altos para todas las métricas, obteniendo pérdidas incluso para la métrica VMAF. Esto sugiere que el deshabilitar el filtro SAO tendría un impacto negativo más pronunciado en este tipo de secuencias en comparación con las demás clases.

A destacar también el caso particular de la métrica VIF, que muestra un comportamiento similar al de VMAF para las secuencias de clase A y B, logrando ahorros promedio de hasta un 0,23 % en BD-Rate.

En general, se observa que deshabilitar el filtro SAO resulta en una pérdida de calidad perceptual para todas las clases (a excepción de la métrica VMAF), siendo esta pérdida más pronunciada en las secuencias de clase F. En cuanto al costo computacional, deshabilitar este filtro no tiene un gran impacto en la complejidad de codificación, reduciendo el tiempo de codificación en un promedio del 0,5 %, según la Tabla 3.9.

3.3.4. Rate-Distortion Optimized Quantization

El algoritmo RDOQ logra un valor de cuantificación óptimo estimado que minimiza el coste Rate-Distortion. Además, para esta evaluación se ha deshabilitado también el

parámetro RDOQTS, el cual desactiva el cálculo RDOQ para los bloques que omiten la etapa de transformación cuando el parámetro TransformSkip se encuentra habilitado.

Tabla 3.6: Rendimiento perceptual promedio al deshabilitar RDOQ [% BD-Rate].

Class	SSIM	MS-SSIM	VMAF	VIF	PSNR HVS-M
A	5,45	2,25	13,37	3,72	5,14
B	4,8	3,49	13,16	4,9	6,2
C	1,81	2,57	10,45	4,29	4,53
D	0,03	0,81	10,77	4,17	4,43
E	2,92	2,34	9,33	3,46	4,43
F	2,08	1,14	6,85	3,3	2,57

La tabla 3.6 refleja el impacto en el rendimiento perceptual promedio al deshabilitar el parámetro RDOQ del estándar HEVC. Aunque la métrica utilizada por el algoritmo RDOQ para medir la distorsión es la PSNR, al observar los resultados vemos que deshabilitar este algoritmo implica un empeoramiento genérico de los valores de BD-Rate para la mayoría de las métricas perceptuales y clases de secuencias de vídeo.

Aunque son todas las métricas las que obtienen pérdidas de calidad perceptual, es la métrica VMAF la que tiene los incrementos más significativos para todas las clases evaluadas, con las clases A y B alcanzando incrementos del 13,37 % y 13,16 % respectivamente. Esto sugiere que el impacto de deshabilitar RDOQ es particularmente perjudicial para las secuencias de vídeo de alta resolución.

La clase D muestra la menor reducción para las métricas basadas la similitud estructural, con valores de 0,03 % para la SSIM y 0,81 % para la MS-SSIM. Sin embargo, las otras métricas, y en mayor medida la VMAF, indican incrementos considerables en BD-Rate para esta clase. Aunque la reducción no es tan drástica como en las clases A y B, deshabilitar RDOQ todavía parece tener un impacto negativo en la calidad del vídeo para las secuencias de muy baja resolución. Las clases C, E y F también muestran incrementos consistentes en todas las métricas.

Con respecto a la complejidad en la codificación, deshabilitar el algoritmo RDOQ implica reducciones importantes en tiempos de codificación para QPs bajas, es decir, cuando se codifica con mayor calidad. A medida que la QP aumenta, el ahorro en tiempos de codificación disminuye, llegando a tener incluso un incremento del 1,5 % para la QP

42, tal y como se muestra en la Tabla 3.9.

Para concluir, es importante destacar que ninguna de las clases muestra un beneficio en términos de calidad perceptual al deshabilitar el algoritmo RDOQ, lo cual indica que este parámetro es crucial para mantener un alto rendimiento perceptual en la codificación.

3.3.5. Transform Skip

Los parámetros TransformSkip y TransformSkipFast están relacionados con la decisión de realizar la etapa de transformación durante el proceso de codificación. Hemos analizado el efecto de deshabilitar ambos parámetros. La tabla 3.7 muestra el impacto en el rendimiento perceptual al deshabilitar ambos parámetros en el estándar HEVC.

Tabla 3.7: Rendimiento perceptual promedio al deshabilitar Transform Skip [% BD-Rate].

Class	SSIM	MS-SSIM	VMAF	VIF	PSNR HVS-M
A	−0,04	−0,04	0,04	−0,04	−0,05
B	−0,07	−0,05	−0,03	−0,07	−0,09
C	−0,21	−0,22	−0,25	−0,16	−0,29
D	−0,07	−0,22	−0,28	−0,16	−0,3
E	−0,07	−0,06	−0,09	−0,06	−0,09
F	4,38	5,32	3,82	7,29	4,57

Para las secuencias con contenido visual real (clases A hasta E), se obtienen ganancias en BD-Rate, pero los valores presentados son muy cercanos a cero, lo que sugiere un cambio mínimo en la calidad perceptual al deshabilitar Transform Skip. Estos valores indican que, para estas clases, deshabilitar el parámetro no tiene un impacto significativo en la calidad del vídeo.

Sin embargo, para la clase F, se observa una notable reducción en la calidad perceptual al deshabilitar Transform Skip, con todos los valores siendo positivos y bastante alejados de cero. La métrica MS-SSIM muestra el mayor incremento de BD-Rate, con un valor de 5,32 %, seguido por la VIF con 7,29 %. Estos valores indican que, para las secuencias de vídeo sintéticas o artificiales, deshabilitar Transform Skip puede tener un efecto muy perjudicial en la calidad perceptual.

En cuanto a la complejidad de codificación, al desactivar Transform Skip, se obtiene un aumento marginal en el tiempo de codificación del 0,2 % en promedio, según la Tabla 3.9.

En términos generales, mientras que deshabilitar Transform Skip no parece tener un impacto significativo en las clases A hasta E, definitivamente afecta negativamente a la clase F. Es crucial considerar la naturaleza y características de las secuencias de vídeo al tomar decisiones sobre si deshabilitar o no este parámetro.

3.3.6. Sign Data Hiding

La tabla 3.8 muestra el impacto en el rendimiento perceptual al deshabilitar el parámetro Sign Data Hiding en el estándar HEVC.

Tabla 3.8: Rendimiento perceptual promedio al deshabilitar SDH [% BD-Rate].

Class	SSIM	MS-SSIM	VMAF	VIF	PSNR HVS-M
A	1,23	1,36	2,7	1,58	1,42
B	1,23	1,21	2,61	1,58	1,27
C	1,01	1,01	2,16	1,19	1,1
D	1,04	1,15	2,16	1,25	1,2
E	0,72	0,76	1,78	1,07	0,92
F	1,15	1,1	1,98	1,15	1,28

Al observar los valores presentados, todos son positivos. Esto indica una disminución general en la calidad perceptual al deshabilitar este parámetro. La métrica VMAF es la que más se ve afectada por la ausencia del algoritmo SDH, con las clases A y B teniendo incrementos en BD-Rate del 2,70 % y 2,61 % respectivamente.

La clase E tiene los valores más bajos entre las métricas, indicando que la calidad perceptual podría no verse tan afectada para las secuencias de esta clase en comparación con las otras. Sin embargo, aún hay una disminución en la calidad percibida.

Podemos concluir que deshabilitar el algoritmo Sign Data Hiding en el estándar HEVC lleva a una reducción en la calidad perceptual en todas las clases analizadas. Aunque el impacto puede variar dependiendo de la clase de secuencia de vídeo, los resultados sugieren que SDH juega un papel importante en la optimización de la calidad perceptual

en la codificación HEVC. Para asegurar una experiencia visual óptima, es recomendable mantener activado este parámetro. Desactivarlo puede llevar a degradaciones notables en la calidad perceptual del vídeo codificado.

3.3.7. Complejidad de codificación

La tabla 3.9 ilustra las variaciones en el tiempo promedio de codificación en CPU [%] al modificar ciertas herramientas de codificación en el estándar HEVC con respecto a su configuración por defecto. Estos cambios se representan en relación con diferentes parámetros de calidad, desde QP 22 (mayor calidad) hasta QP 42 (menor calidad).

Tabla 3.9: a

l variar el estado de las herramientas de codificación con respecto a su estado predeterminado]Aumento/disminución media relativa de tiempos de codificación [%] al variar el estado de las herramientas de codificación con respecto a su estado predeterminado. Valores negativos indican ahorros en tiempos de codificación).

	QP 22	QP 27	QP 32	QP 37	QP 42	Promedio
SCL habilitado	7,41	8,44	6,57	5,18	3,47	6,21
SAO deshabilitado	−0,3	−0,21	−0,53	−0,37	−0,58	−0,4
DB deshabilitado	−0,22	−0,18	−0,41	−0,24	−0,26	−0,26
RDOQ deshabilitado	−16,56	−10,77	−6,4	−1,89	1,5	−6,82
TrSkp deshabilitado	−15,78	−14,83	−14,24	−13,29	−13,22	−14,27
SDH deshabilitado	−3,25	−2,72	−2,17	−1,49	−1,03	−2,13

Lo primero que destaca es el resultado de habilitar el parámetro Scaling List. Esta herramienta, cuando está habilitada, aumenta significativamente el tiempo de codificación en todas las QPs, siendo el impacto más notable en QP 22 con un incremento del 7,41 %. A medida que la calidad de codificación disminuye (hacia QP 42), el incremento en el tiempo de codificación también se reduce, aunque sigue siendo elevado. En promedio, habilitar la Scaling List conduce a un aumento del 6,21 % en el tiempo de codificación.

Por otro lado, deshabilitar las otras herramientas produce ahorros en el tiempo de codificación. Entre estas, la deshabilitación del algoritmo Rate Distortion Optimized Quantization y Transform Skip son las que más influyen en el tiempo de codificación, mostrando reducciones sustanciales. Deshabilitar RDOQ lleva a ahorros notables, especialmente en las calidades más altas (QP 22 y 27) con reducciones de −16,56 % y −10,77 % respectivamente. Esta tendencia cambia en QPs altas (baja calidad), donde incluso se observa un ligero incremento del tiempo de codificación al deshabilitar RDOQ. Sin embargo, en pro-

medio, deshabilitar RDOQ produce un ahorro del 6,82 %. De manera similar, deshabilitar Transform Skip conlleva a reducciones consistentes en el tiempo de codificación a través de todas las QPs, con un ahorro promedio del 14,27 %.

Las otras herramientas, aunque también producen ahorros en tiempo, tienen un impacto menor en comparación con RDOQ y Transform Skip. Deshabilitar el filtro SAO, el filtro de Deblocking y Sign Data Hiding muestra ahorros que, aunque consistentes, son más modestos en magnitud, siendo el algoritmo SDH el tiene un mayor impacto en tiempos de cómputo, con un 2,13 % en promedio.

El tiempo de codificación en HEVC se ve considerablemente afectado por la habilitación o deshabilitación de ciertas herramientas. Es esencial ponderar el equilibrio entre calidad y eficiencia, ya que cada herramienta tiene un impacto único en el proceso. Mientras que la habilitación de Scaling List aumenta el tiempo de codificación, herramientas como RDOQ y Transform Skip, cuando se deshabilitan, ofrecen ahorros significativos en tiempo. La elección de habilitar o deshabilitar estas herramientas debe hacerse con base en las necesidades específicas de cada caso y de los recursos disponibles, considerando siempre la relación entre la calidad perceptual y los tiempos de codificación.

3.4. Discusión

En el apartado anterior se ha comprobado que la mayoría de las herramientas de codificación presentan una respuesta o comportamiento perceptual variable al modificar su estado predeterminado, ya sea entre las diferentes métricas utilizadas o entre las distintas clases secuencias de vídeo evaluadas. En esta sección se va a analizar y discutir en profundidad estos resultados, considerando además la interrelación de estados entre las diferentes herramientas de codificación.

Se ha demostrado que mantener habilitado el parámetro Transform Skip es efectivo únicamente para las secuencias de clase F. Para el resto de las clases, deshabilitarlo incrementa levemente la respuesta perceptual para todas las métricas, obteniendo la mayor ganancia en las clases de media y baja resolución (C y D respectivamente). Teniendo en cuenta la enorme reducción de tiempos de codificación cuando se deshabilita esta herramienta de codificación, perceptualmente se recomienda deshabilitar su uso a excepción

de los casos donde se codifiquen secuencias de vídeo sintéticas o generadas por ordenador, en cuyo caso se deberá evaluar el coste computacional frente a la ganancia perceptual obtenida.

En cuanto a la herramienta de codificación Scaling List (SCL), se reporta una mejor respuesta perceptual por casi todas las métricas, con independencia del estado del resto de herramientas de codificación. Al habilitar la cuantificación perceptual, se logran ahorros promedio de BD-Rate de hasta un 2,65 %, siendo el promedio de todas las clases y métricas de prácticamente un 1 %. La herramienta SCL actúa conforme a lo esperado, pues está demostrado que el uso de un cuantificador basado en CSF, como el implementado en el estándar HEVC a través de esta herramienta, mejora la calidad subjetiva del vídeo decodificado.

Al deshabilitar el algoritmo RDOQ, todas las métricas muestran incrementos significativos en BD-Rate para todas las clases de vídeo. Se observa un incremento promedio de BD-Rate del 4,83 %, aunque este incremento aumenta hasta el 10,65 % únicamente para la métrica VMAF, que perceptualmente es la mayor perjudicada al deshabilitar esta herramienta de codificación. Por lo tanto, sugerimos no desactivar el parámetro RDOQ. No obstante, como toda regla, hay una excepción. En el caso de la métrica SSIM, se percibe una reducción promedio del BD-Rate del 0,08 % en las secuencias de la clase D.

Nuestro análisis revela que, para ofrecer un mejor rendimiento de calidad perceptual para todas las métricas objetivas y clases de secuencias de vídeo, deberíamos (a) activar las herramientas de codificación SCL y RDOQ, y (b) activar TrSk solo cuando trabajemos con secuencias de vídeo de la clase F.

Basándonos en esta consideración, examinaremos el comportamiento de los filtros in-loop. Se observa que el comportamiento general de los filtros in-loop es algo contradictorio: por un lado, las métricas SSIM, MS-SSIM y PSNR-HVS-M proporcionan los mejores resultados perceptuales en todas las clases de vídeo cuando ambos filtros están activados. Por otro lado, VMAF y VIF indican lo contrario, mostrando los mayores ahorros en BD-Rate cuando ambos filtros están desactivados. Sin embargo, es importante mencionar que existe un consenso entre todas las métricas para activar ambos filtros in-loop y maximizar el ahorro de BD-Rate al trabajar con vídeos de la clase F.

A pesar de que todas las métricas de calidad objetiva están diseñadas para evaluar la

calidad de una manera cercana a cómo lo hace el sistema visual humano (HVS), cada una utiliza una aproximación diferente, lo que resulta en variadas evaluaciones de calidad dependiendo de la métrica utilizada. En casos donde todas las métricas reportan variaciones de BD-Rate en la misma dirección, la conclusión es clara, pero cuando los informes de las métricas son opuestos, se sugiere una prueba subjetiva para validar los resultados.

Para proporcionar una evaluación subjetiva preliminar que clarifique la controversia de la métrica VMAF respecto al uso de los filtros in-loop, se ha realizado una prueba subjetiva con una secuencia de vídeo de clase A, escogiendo un fotograma donde la ganancia (siempre según la VMAF) entre deshabilitar ambos filtros in-loop es máxima. El resultado de dicha prueba muestra que, a una QP elevada (32 y 37), se observan claramente los artefactos propios de la codificación basada en bloques cuando se deshabilitan los filtros in-loop (ver la Figura 3.6). Por tanto, podemos concluir que la métrica VMAF no está funcionando correctamente en estos casos, ya que devuelve un valor de ganancia elevado al deshabilitar estos filtros.

Es esencial mencionar que los resultados proporcionados por el VMAF no están sesgados por el contenido de la imagen o el tamaño del fotograma, demostrando consistencia a través de diferentes tamaños de fotograma y contenido. Se ha reportado una buena correlación con los valores DMOS y MOS del VMAF, indicando que puede considerarse una métrica robusta. A pesar de que los resultados en la Figura 3.6 parecen indicar que la métrica VMAF no evalúa correctamente la calidad percibida cuando ambos filtros están desactivados, muestra buenos resultados cuando solo el filtro DB está activado.

Finalmente, otra métrica de rendimiento que podríamos utilizar para evaluar la configuración más adecuada de las herramientas de codificación es su contribución a la complejidad global de codificación HEVC. En la Tabla 3.9, hemos presentado los resultados de tiempo de cada herramienta de codificación individual en estudio, mostrando su impacto en la complejidad total de codificación HEVC.

(a) SAO y DB habilitados.

(b) SAO deshabilitado, DB habilitado.

(c) SAO habilitado, DB deshabilitado.

(d) SAO y DB deshabilitados.

Figura 3.6: Comparativa de un recorte del fotograma 22 de la secuencia Traffic (2560x1660) codificado con QP = 42, a 8,5$Mbps$. En (a), aparece la codificación por defecto (filtros in-loop habilitados), mientras que en el resto de los casos se ha deshabilitado uno o ambos filtros.

3.5. Conclusiones

Este estudio ha explorado cómo diversas herramientas de codificación en el estándar HEVC impactan el Rate-Distortion desde una perspectiva perceptual. Utilizando un conjunto amplio de secuencias de vídeo y métricas perceptuales, hemos identificado configuraciones clave que maximizan el rendimiento perceptual.

Hemos encontrado que las herramientas SCL, RDOQ y SBH son fundamentales para mantener un alto rendimiento de calidad perceptual y, por lo tanto, deben estar siempre habilitadas. Por otro lado, la herramienta Transform Skip es beneficiosa únicamente para vídeos de clase F, mientras que, para otros tipos, su desactivación puede mejorar ligeramente la respuesta perceptual y reducir significativamente los tiempos de codificación.

En cuanto a los filtros in-loop, SAO y DB, hemos observado comportamientos divergentes. Si bien las métricas SSIM, MS-SSIM y PSNR-HVS-M sugieren que ambos filtros

deben estar activados para maximizar los ahorros de BD-Rate, la métrica VMAF y VIF aconsejan lo contrario. Este estudio revela la necesidad de realizar pruebas subjetivas para cada caso específico debido a la falta de consenso en los resultados.

Además, se ha demostrado que deshabilitar RDOQ conduce a un incremento notable en el BD-Rate, lo que sugiere siempre su activación. La evaluación subjetiva preliminar indica que VMAF puede no evaluar adecuadamente la calidad percibida cuando ambos filtros in-loop están desactivados, a pesar de su buena correlación con los valores DMOS y MOS.

Este trabajo contribuye a la comprensión del impacto de las herramientas de codificación en el HEVC, proporcionando guías para seleccionar la configuración del codificador adecuada según el tipo de secuencia y el objetivo de maximizar el rendimiento de la tasa de distorsión perceptual. Futuras investigaciones incluirán la ampliación del estudio a más herramientas de codificación HEVC y la realización de pruebas subjetivas exhaustivas para determinar configuraciones basadas en la percepción.

4. MODELO HÍBRIDO DE ENMASCARAMIENTO POR CONTRASTE Y TEXTURA PARA MEJORAR EL RENDIMIENTO PERCEPTUAL DE HEVC

En el contexto de la compresión de vídeo, es esencial que las técnicas de codificación no solo reduzcan el tamaño de los archivos, sino que también mantengan una alta calidad perceptual para el usuario final. Este estudio se centra en dos técnicas de codificación perceptual, el enmascaramiento por contraste y por textura, que operan conjuntamente en el estándar de codificación de vídeo de alta eficiencia (HEVC). Dichas técnicas explotan características del Sistema Visual Humano (HVS) para enmascarar artefactos de compresión y mejorar la calidad subjetiva del vídeo reconstruido, sin aumentar significativamente la tasa de bits.

En este estudio se presenta un esquema novedoso que combina técnicas de enmascaramiento por contraste y textura aplicadas al software de referencia del estándar HEVC. Para el enmascaramiento por contraste, se utilizan matrices de cuantificación dependientes de la frecuencia para bloques de tamaños desde 8×8 hasta 32×32, introduciendo una nueva matriz de pesos para los bloques de tamaño 4×4, permitiendo una mayor compresión sin comprometer la calidad perceptual. En el caso del enmascaramiento de textura, se emplea la métrica de la Varianza Direccional Media (MDV) [28] y un modelo de Máquina de Vectores Soporte (SVM) para clasificar los bloques según tu nivel de textura. A partir de esta clasificación, se ajusta el valor del QP (QP offset) en función de la energía de cada bloque.

Las principales contribuciones de este trabajo incluyen un método mejorado de enmascaramiento de contraste que cubre todos los tamaños de bloque en HEVC, un nuevo método de clasificación de bloques basado en la métrica MDV que clasifica eficientemente los bloques según su tipo de textura, y un nuevo calculador de QP offset para el sistema de QP adaptativo de HEVC, basado en la energía de la textura y su clasificación.

4.1. Metodología propuesta

En esta sección se detallan los aspectos clave del nuevo esquema de cuantificador perceptual propuesto. Primero, se describe cómo se aplica el enmascaramiento por contraste a los bloques de tamaño 4×4, y las mejoras propuestas en este enfoque. Posteriormente, tras la aplicación del enmascaramiento por contraste, se introduce un esquema de sobre cuantificación basado en el enmascaramiento de textura, que depende de (a) un nuevo clasificador de bloques y (b) un cuantificador optimizado en función del tipo de bloque y su energía.

4.1.1. Matriz de cuantificación 4×4 basada en un modelo de sensibilidad al contraste

El estándar HEVC soporta matrices de cuantificación dependientes de la frecuencia para bloques de tamaño 8×8, que se extrapolan para bloques más grandes (ver Figura 2.15). Esta herramienta, denominada *ScalingList*, sin embargo, no incluye matrices específicas para bloques de tamaño 4×4, aplicándose siempre una matriz uniforme (ver Apartado 3.1.1).

En este trabajo, proponemos una matriz de cuantificación no uniforme para bloques de tamaño 4×4, diseñada específicamente para aumentar la compresión en los bloques más pequeños sin comprometer la calidad perceptual. A diferencia del enfoque de sobre muestreo habitual en el estándar, basamos los pesos de la matriz en el estudio presentado por Onofre [63], que propone el uso del modelo de sensibilidad al contraste de Daly (Ecuación 4.1), donde f representa la frecuencia radial en ciclos por grado (cpd), asumiendo condiciones de visualización donde los defectos se detectan de forma temprana, como el uso de pantallas de alta resolución y distancias de visualización cortas.

$$H(f) = 2{,}2(0{,}192 + 0{,}114 \cdot f) \cdot e^{-(0{,}114 \cdot f)^{1{,}1}} \tag{4.1}$$

La ecuación de Daly, que actúa como un filtro paso-banda, es más sensible en frecuencias intermedias y menos en frecuencias bajas y altas. Saturamos las frecuencias por debajo del pico de sensibilidad para preservar la información cercana a la componente discreta (DC), donde se concentra la mayor parte de la energía tras aplicar la Transformada Discreta del Coseno (DCT) a un bloque.

Para determinar la frecuencia máxima (f_{max}), calculamos primero la frecuencia de muestreo (f_s) con la Ecuación 4.2, basada en una resolución de $r = 600$ píxeles por pulgada (ppi) y una distancia de visualización $\upsilon = 12{,}23$ pulgadas. Aplicando el teorema de Nyquist-Shannon, la frecuencia máxima es la mitad de f_s (Ecuación 4.3), obteniendo $f_{max} = 64{,}04$ cpd.

$$f_s = \frac{\upsilon \cdot \tan(1°) \cdot r}{0{,}0254} \tag{4.2}$$

$$f_{max} = \frac{f_s}{2} \tag{4.3}$$

La Figura 4.1 muestra la curva CSF obtenida con la Ecuación 4.1. La curva azul representa la versión modificada, donde se saturan las bajas frecuencias para preservar la información cercana al DC.

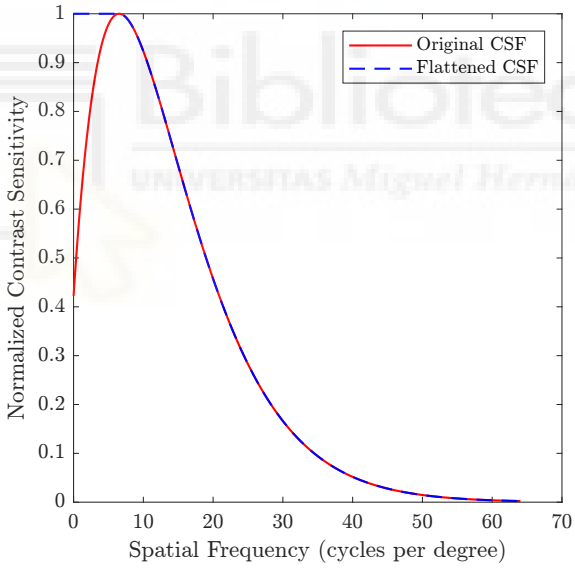


Figura 4.1: Función de sensibilidad al contraste. La curva roja representa la CSF original según Daly, y la azul discontinua muestra la CSF modificada.

Si observamos la curva roja de la Figura 4.1, podemos comprobar cómo el sistema visual humano es más sensible en una región frecuencial intermedia, actuando como un filtro paso banda, mientras que es menos sensible a frecuencias muy bajas y muy altas. La curva azul muestra la saturación de las bajas frecuencias por debajo del nivel de sensibilidad máxima. Esto se hace para preservar la información de los coeficientes cercanos

a la componente continua, incluyendo esta, ya que es en esa región donde se concentra la mayor parte de la información o energía tras aplicar la DCT a un bloque.

Para construir la matriz 4×4, calculamos la frecuencia radial $f(u, v)$ para cada coeficiente DCT utilizando la Ecuación 4.4, donde $u, v \in 0, 1, 2, 3$ representan las coordenadas del bloque. Estas frecuencias se mapean en la curva CSF para obtener los valores de sensibilidad, que luego se normalizan y escalan para ajustarse al estándar HEVC.

$$f(u, v) = \sqrt{u^2 + v^2} \tag{4.4}$$

El resultado final son las matrices de ponderación propuestas (ver Figura 4.2) para los modos de predicción Intra e Inter.

$$\begin{bmatrix} 16 & 16 & 20 & 32 \\ 16 & 17 & 21 & 37 \\ 20 & 21 & 29 & 55 \\ 32 & 37 & 55 & 115 \end{bmatrix} \quad \begin{bmatrix} 16 & 16 & 19 & 29 \\ 16 & 17 & 20 & 32 \\ 19 & 20 & 26 & 46 \\ 29 & 32 & 46 & 91 \end{bmatrix}$$

(**a**) Intra-predicción (**b**) Inter-predicción

Figura 4.2: Propuesta de matrices de cuantificación no uniformes para bloques 4×4 en modos (a) Intra e (b) Inter.

Para evaluar el impacto de perceptual de nuestra propuesta, hemos codificado las secuencias de testeo (ver Tabla 3.1), utilizando las matrices predefinidas en el estándar (SCL=1), frente a nuestra propuesta para bloques 4×4 (SCL=2). Se ha deshabilitado la herramienta de codificación *TransformSkip* para todas las secuencias excepto las de clase F, para maximizar la respuesta perceptual tal y como recoge el Apartado 3.3.5 y [10].

Los resultados de la Tabla 4.1 muestran que nuestra propuesta ofrece mejoras significativas en todas las clases de secuencias y métricas perceptuales en comparación con la configuración por defecto, destacando la necesidad de incluir esta matriz de 4×4 en el estándar.

Tabla 4.1: Rendimiento perceptual promedio [% BD-rate] al habilitar las matrices de cuantificación no uniformes.

Secuencia / Clase	SCL=1 (por defecto)			SCL=2 (nosotros)		
	SSIM	MS-SSIM	PSNR-HVS-M	SSIM	MS-SSIM	PSNR-HVS-M
Clase A	−0,66	−0,33	−0,62	−1,06	−0,82	−1,58
Clase B	−0,97	−0,48	−0,99	−3,20	−2,58	−4,23
Clase C	0,26	0,08	−0,08	−4,82	−5,36	−7,39
Clase D	1,26	0,29	−0,05	−1,36	−5,66	−7,65
Clase E	−0,74	−0,50	−0,75	−1,78	−1,39	−1,98
Clase F	−0,15	−0,04	−0,11	−4,57	−4,19	−4,17

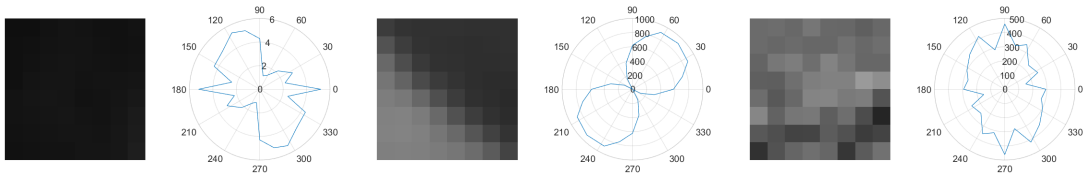
4.1.2. Clasificador de bloques usando SVM basado en la métrica MDV

Una vez aplicado el enmascaramiento por contraste, calculamos el QP offset en función del nivel de textura del bloque. Para ello, utilizamos un clasificador de bloques similar al propuesto por Tong *et al.* [26] para JPEG, quienes propusieron clasificar los bloques en planos (Plain), de borde (Edge) y con textura (Texture). En su propuesta, los bloques planos deben evitar sobre cuantificación, los de borde pueden ser levemente sobre cuantificados, y los de textura pueden serlo en mayor medida según su nivel de energía.

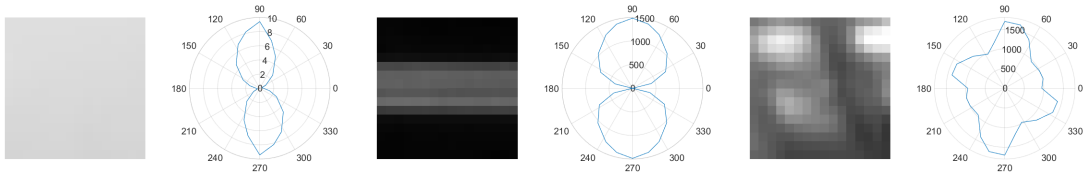
El problema de adaptar este esquema a HEVC radica en la variedad de tamaños de bloque que utiliza el estándar, en contraste con el JPEG, que usa solo bloques de 8×8. Además, HEVC emplea transformadas enteras (I-DCT e I-DST) aplicadas al error residual de las predicciones. Por eso, proponemos un clasificador de bloques supervisado mediante una SVM, utilizando la métrica *Mean Directional Variance* (MDV) definida por Damián *et al.* [28] como características de entrada.

Primero, clasificamos manualmente unos 1800 bloques de luminancia en HEVC, de distintos tamaños y con diferente nivel de textura. Usamos bases de datos como ESPL [90], USC-SIPI [91], TESTIMAGE [92] y Kodak [93] para obtener bloques de forma aleatoria. Se programó un pequeño software en MATLAB para externalizar y distribuir la tarea de clasificación entre diferentes investigadores, con el fin de evitar sesgos en la clasificación.

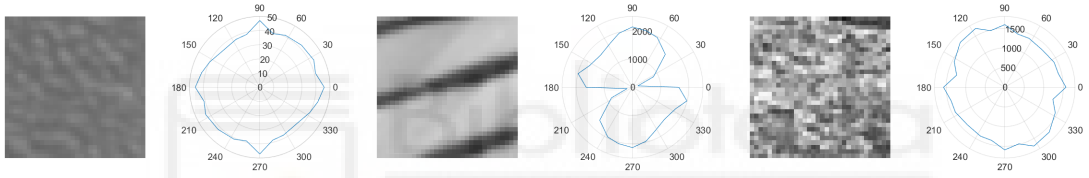
La Figura 4.3 muestra algunos de los bloques clasificados manualmente. Los bloques planos tienen contenido suave, los de textura un patrón más aleatorio, y los de borde una direccionalidad muy pronunciada.



(a) Bloque plano 8×8 (b) Bloque de borde 8×8 (c) Bloque de textura 8×8



(d) Bloque plano 16×16 (e) Bloque de borde 16×16 (f) Bloque de textura 16×16



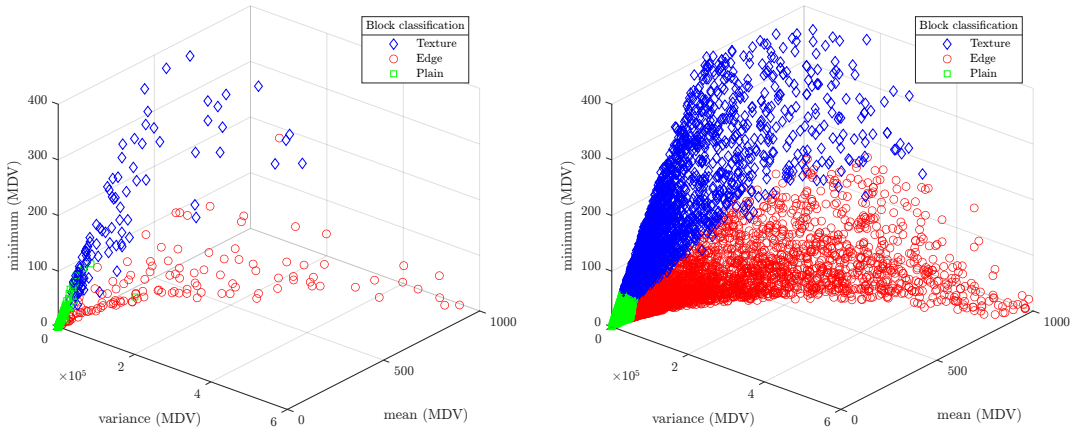
(g) Bloque plano 32×32 (h) Bloque de borde 32×32 (i) Bloque de textura 32×32

Figura 4.3: Ejemplos de bloques clasificados manualmente (izquierda) y su diagrama polar asociado de la métrica MDV (derecha).

La Figura 4.3 también muestra el diagrama polar de MDV para cada bloque. Esta métrica mide la varianza acumulada a lo largo de diferentes direcciones. Los bloques de textura presentan un patrón circular en el diagrama polar, mientras que los bloques de borde tienen una marcada orientación (un mínimo) en la dirección del borde. Los bloques planos muestran valores bajos de MDV, indicando contenido suave.

Para una clasificación precisa de bloques, empleamos un modelo de SVM desarrollado en MATLAB. Dado que queríamos tres grupos (plano, borde y textura), utilizamos las estrategias multiclase *One-vs-One* y *One-vs-Rest*, obteniendo resultados similares en términos de coste computacional, dado que se requieren el mismo número de modelos binarios en ambos casos. Usamos estadísticas como la media, varianza y el valor mínimo de MDV para entrenar el clasificador.

Obtenidos los modelos SVM para cada tamaño de bloque, estos lograron una precisión alta en la clasificación (93,9 %, 95,4 % y 94,5 % para bloques 8×8, 16×16 y 32×32, respectivamente). La Figura 4.4a muestra los resultados de entrenamiento, donde los bloques planos se agrupan cerca del origen y los bloques de textura y borde ocupan diferentes planos.



(a) Conjunto de datos de entrenamiento

(b) Conjunto de datos de testeo

Figura 4.4: Diagrama de dispersión de bloques 16×16 para (a) conjunto de datos de entrenamiento y (b) conjunto de datos de testeo.

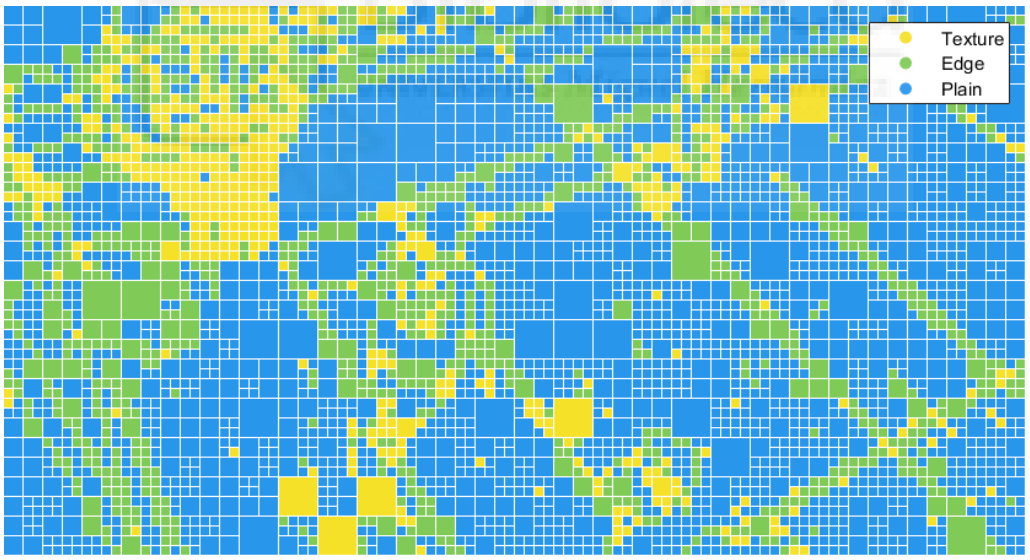
Finalmente, exportamos los modelos SVM de MATLAB a C++ e integramos la clasificación de bloques en el codificador HM de HEVC. Esta clasificación de bloques se realiza antes de la etapa de particionado y RDO, almacenándose el tipo de bloque y su energía (ε) para su posterior uso en la etapa de cuantificación.

La Figura 4.5 muestra un ejemplo de clasificación en la secuencia *BasketballDrill* a QP 32. Las líneas de la cancha se clasificaron como bloques de borde, mientras que partes de la red fueron clasificadas como textura.

4.1.3. Obtención del QP offset óptimo

El siguiente paso después de clasificar un bloque CU es obtener su QP offset óptimo. Para ello, definimos la energía del bloque (ε) como la suma absoluta de todos los coeficientes transformados AC. Se analizó esta energía según el tipo de bloque (textura, borde o plano) y su tamaño. La Figura 4.6 muestra la distribución de la energía en un diagrama de caja, permitiendo visualizar el resumen de cinco números: los valores mínimo y

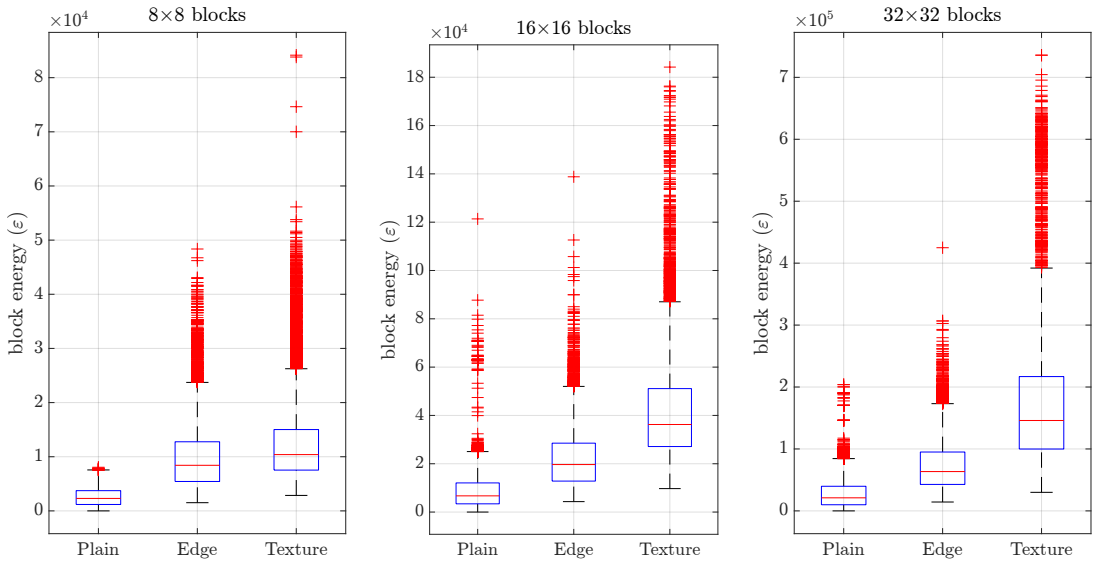
(a) Fotograma original de BasketballDrill.



(b) Clasificación de bloques utilizando QP=32.

Figura 4.5: Clasificación de bloques para el primer fotograma de BasketballDrill usando los modelos SVM para cada tamaño de bloque.

máximo, los cuartiles superior e inferior, y la mediana.



(a) Tamaño de bloque 8×8 (b) Tamaño de bloque 16×16 (c) Tamaño de bloque 32×32

Figura 4.6: Diagrama de caja de la distribución de energía del bloque (ε) por tamaño y clasificación de textura.

Los bloques de textura tienen la mayor energía, seguidos de los bloques de borde y los bloques planos, como se esperaba. Los valores atípicos (outliers) que se observan en la Figura 4.6 provienen de secuencias sintéticas generadas por ordenador, con alta energía en bandas medias y altas, lo que los diferencia de los bloques de secuencias naturales.

Para calcular el QP offset (ΔQP) de cada CU, seguimos un enfoque basado en la energía del bloque, utilizando métricas de distorsión perceptual como SSIM, MS-SSIM o PSNR-HVS-M. La Ecuación 4.5, basada en [57], muestra cómo calcular ΔQP a partir del tamaño del paso de cuantificación $QStep_{i,j}$. Si $QStep_{i,j} = 1$, no se aplica cuantificación adicional al bloque.

$$\Delta QP_{i,j} = \left\lfloor \frac{6 \cdot \ln(QStep_{i,j})}{\ln(2)} \right\rfloor \tag{4.5}$$

La función lineal para obtener $QStep_{i,j}$, mostrada en la Ecuación 4.6, sigue un modelo similar al propuesto en [26], [27], [74], donde $MinE$ y $MaxE$ representan la energía mínima y máxima para un conjunto de bloques de igual tamaño y tipo, y $MaxQStep$ es el paso máximo de cuantificación permitido.

$$QStep_{i,j} = \begin{cases} 1 & \text{if } \varepsilon(B_{i,j}) \leq MinE, \\ MaxQStep & \text{if } \varepsilon(B_{i,j}) \geq MaxE, \\ 1 + \frac{MaxQStep-1}{MaxE-MinE} \times & \\ (\varepsilon(B_{i,j}) - MinE) & \text{otherwise} \end{cases} \tag{4.6}$$

Se probaron diferentes conjuntos de parámetros para cada tamaño (8×8, 16×16 y 32×32) y tipo de bloque (textura y borde), como se resume en la Figura 4.7. No se consideraron los bloques planos debido a su alta sensibilidad frente a artefactos [26]. La selección de parámetros se basó en el análisis por fuerza bruta, restringiendo $MaxQStep$ entre 1,1 y 2,5 para asegurar que $\Delta QP_{i,j}$ estuviera en el rango máximo permitido por el estándar HEVC.

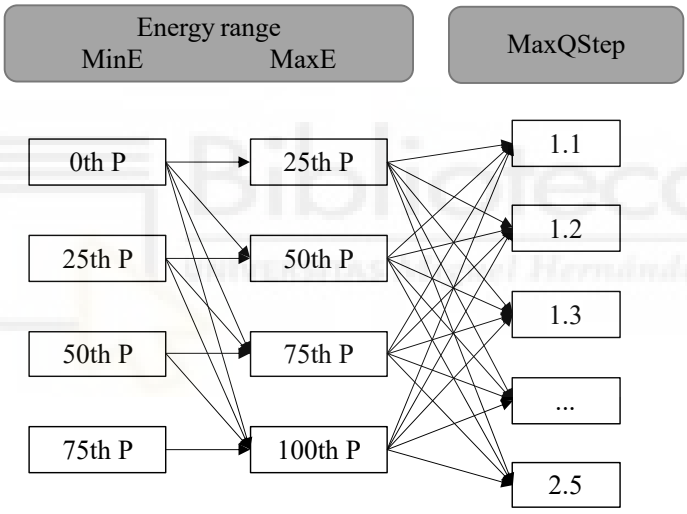


Figura 4.7: Diagrama de flujo de la selección de candidatos para el análisis por fuerza bruta de parámetros óptimos perceptuales. Las P en las cajas de rango de energía se refieren al percentil.

Se encontraron parámetros óptimos que definen la función lineal para cada tipo y tamaño de bloque. Posteriormente, se realizó un conjunto de codificaciones con diferentes valores de QP (22, 27, 32 y 37) usando secuencias de testeo de alta resolución (clases A, B y E) [84].

Como ejemplo, la Figura 4.8 muestra la ganancia obtenida en BD-Rate (MS-SSIM) para los bloques de textura de tamaño 8×8 en la secuencia PeopleOnStreet. La mayor ganancia se produce cuando $MaxElevation = 1{,}3$, para un rango de energía entre el

percentil 0 y 25.

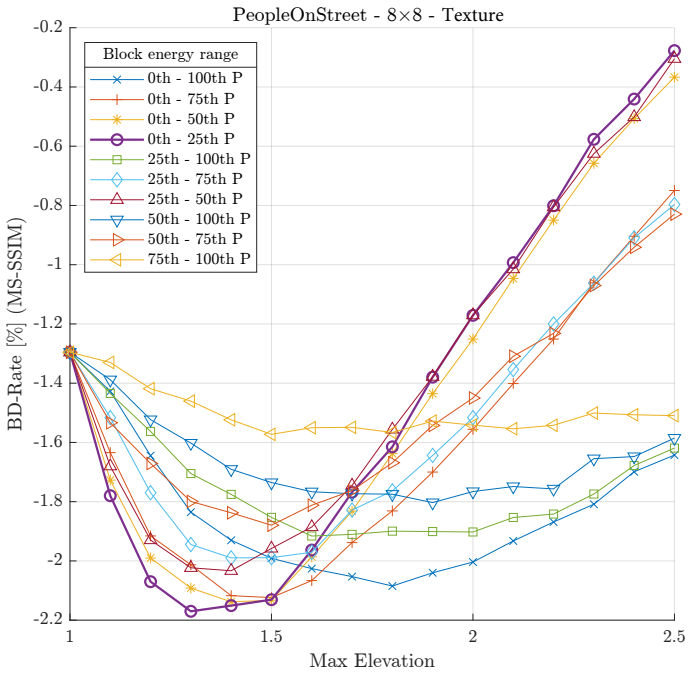


Figura 4.8: Curvas BD-Rate (métrica MS-SSIM) para la secuencia de prueba de vídeo PeopleOnStreet sobre el parámetro $MaxQStep$ al modificar bloques de textura de tamaño 8. Cada curva representa un rango de energía de bloque diferente ($MinE$ y $MaxE$).

Nuestra propuesta de enmascaramiento de textura consiguió importantes incrementos de ganancia en BD-Rate, lo que demuestra la eficacia del enfoque en términos de métricas perceptuales.

4.2. Resultados y Discusión

Para evaluar el comportamiento global de nuestra propuesta de cuantificador perceptual en HEVC, realizamos una exhaustiva evaluación de los modelos de enmascaramiento por contraste y textura descritos anteriormente. Basándonos en las recomendaciones de las pruebas de conformidad de HEVC [84], utilizamos todas las secuencias de testeo agrupadas por clases (ver Tabla 3.1) y la métrica BD-rate [7], aplicando SSIM, MS-SSIM y PSNR-HVS-M como métricas de calidad perceptual. Los valores base de QP utilizados fueron 22, 27, 32 y 37.

Nuestra propuesta de enmascaramiento por contraste y textura fue implementada en la versión 16.20 del software de referencia HEVC [94], ejecutada en un servidor Linux de

alto rendimiento con CPUs Intel® Xeon® Gold 6140 y 376 GB de RAM. Para asegurar la compatibilidad con el estándar HEVC, señalizamos los valores de QP offset a nivel de CU, ya que HEVC permite la transmisión de un ΔQP para cada bloque CU [57].

En la Tabla 4.2, se presentan los resultados para las configuraciones de codificación All Intra (AI), Random Access (RA) y Low Delay (LD). La columna "Masking por contraste" muestra las ganancias obtenidas con nuestro modelo de enmascaramiento por contraste (Apartado 4.1.1), mientras que la columna "Masking por contraste y textura" refleja las ganancias al aplicar tanto el enmascaramiento por contraste como el de textura (Apartado 4.1.2).

Como se puede comprobar, el uso de ambas técnicas propuestas en este estudio resultó siempre en mayores reducciones de BD-Rate en comparación con solamente el uso del enmascaramiento por contraste. Por otro lado, no hay diferencias significativas de las ganancias perceptuales si comparamos las diferentes clases entre los tres modos de predicción utilizados, lo cual es un indicativo de que nuestra propuesta mejora también las predicciones Inter.

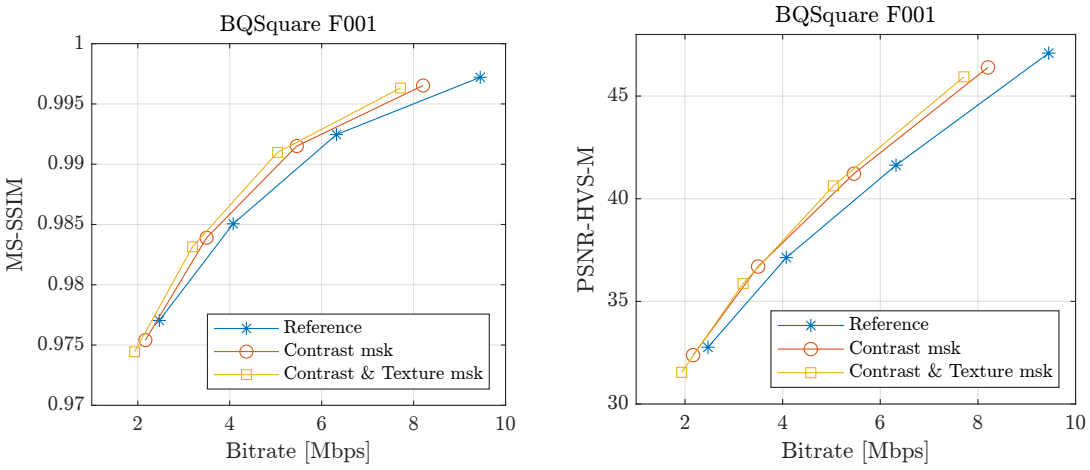
Las reducciones de BD-Rate fueron especialmente significativas para las clases de resolución media y baja (C y D), con ganancias promedio de hasta $-12{,}82\,\%$, mientras que las clases de mayor resolución (A y E) presentaron menores ganancias, entre $-1{,}65\,\%$ y $-4{,}01\,\%$. En promedio, se obtuvieron ahorros en BD-Rate superiores al $5\,\%$, con ganancias máximas de hasta $12{,}89\,\%$.

Como ejemplo, en la Figura 4.9, se muestran las curvas R/D del primer fotograma de la secuencia BQSquare, donde nuestra propuesta mejora el rendimiento perceptual en todas las métricas evaluadas.

Además, como se observa en la Figura 4.9, los mayores ahorros de tasa de bits se lograron en bajas tasas de compresión. Para un QP de 22, se obtuvo un ahorro de bits del $18{,}3\,\%$, al pasar de 9,45 Mbps en la codificación por defecto a 7,72 Mbps utilizando enmascaramiento por contraste y textura.

Tabla 4.2: Rendimiento promedio de codificación en diferentes configuraciones [% BD-Rate].

Configuración	Clase	Masking por contraste			Masking por contraste y textura		
		SSIM	MS-SSIM	PSNR-HVS-M	SSIM	MS-SSIM	PSNR-HVS-M
All Intra	A	−1.06	−0.82	−1.58	−2.02	−1.65	−1.70
	B	−3.20	−2.58	−4.23	−4.82	−3.92	−4.79
	C	−4.82	−5.36	−7.26	−7.09	−7.58	−8.31
	D	−1.36	−5.66	−7.65	−3.41	−8.68	−8.98
	E	−1.78	−1.39	−1.98	−3.38	−2.79	−2.27
	F	−4.57	−4.19	−4.17	−7.52	−6.98	−5.78
Promedio AI		−2.80	−3.33	−4.48	−4.71	−5.27	−5.30
Random-Access	A	−1.42	−1.01	−1.55	−3.71	−3.15	−2.50
	B	−4.30	−3.66	−5.37	−6.67	−5.77	−6.40
	C	−3.77	−4.01	−6.20	−7.43	−7.61	−8.26
	D	0.02	−5.02	−7.20	−4.12	−9.65	−9.64
	E	−1.90	−1.46	−2.15	−4.01	−3.32	−3.13
	F	−3.64	−3.31	−3.76	−7.32	−6.94	−6.08
Promedio RA		−2.50	−3.08	−4.37	−5.54	−6.08	−6.00
Low-Delay	A	−1.26	−0.90	−1.50	−3.51	−3.05	−2.56
	B	−3.56	−3.06	−4.92	−6.16	−5.51	−6.34
	C	−4.03	−4.22	−6.38	−8.35	−8.53	−8.99
	D	−3.56	−7.04	−9.04	−8.57	−12.89	−12.35
	E	−0.71	−0.41	−1.16	−2.95	−2.44	−1.93
	F	−3.62	−3.52	−3.45	−7.82	−7.43	−5.73
Promedio LD		−2.79	−3.19	−4.41	−6.23	−6.64	−6.32



(a) (b)

Figura 4.9: Comparativa de curvas R/D de BQSquare, utilizando las métricas perceptuales (**a**) MS-SSIM y (**c**) PSNR-HVS-M.

4.3. Conclusiones

Las técnicas de compresión basadas en el sistema visual humano (HVS), como el enmascaramiento de textura y contraste, han demostrado durante años su capacidad para reducir la tasa de bits sin afectar la calidad visual. En este trabajo, hemos desarrollado un esquema novedoso que combina de forma eficiente ambas técnicas en el software de referencia del estándar HEVC, mostrando su capacidad para reducir la tasa de bits manteniendo una calidad perceptual similar.

La incorporación de nuestra matriz de cuantificación no uniforme 4×4 permitió una reducción promedio del BD-Rate entre un 2,69 % (SSIM) y un 4,42 % (PSNR-HVS-M) en todas las secuencias de testeo y modos de codificación.

Además, desarrollamos un algoritmo de clasificación de bloques utilizando la varianza direccional media y una máquina de soporte vectorial, lo que dio lugar a un modelo de enmascaramiento por textura que, junto con el enmascaramiento por contraste, logró una reducción promedio total de BD-Rate entre un 5,49 % (SSIM) y un 5,99 % (MS-SSIM).

En trabajos futuros, exploraremos la posibilidad de aplicar la sobre cuantificación para bloques de 4×4 en el software de referencia HEVC, con el objetivo de mejorar aún más el rendimiento de nuestro modelo de enmascaramiento de textura. También investigaremos un sistema de preprocesamiento que permita identificar las secuencias donde el enmascaramiento no aporta beneficios perceptuales. Por último, evaluaremos otras técnicas de codificación perceptual, como el enmascaramiento por luminancia o el uso de métricas de atención, que podrían complementar y mejorar los resultados obtenidos.

5. CONCLUSIONES

Esta tesis doctoral ha abordado con éxito el estudio y aplicación de técnicas de codificación perceptuales para mejorar la eficiencia de compresión y la calidad visual de los contenidos de vídeo en plataformas basadas en IP. El objetivo principal se ha centrado en la implementación y evaluación de métodos que mejoren la percepción visual de vídeos en HD/UHD, fundamentado por una serie de investigaciones exhaustivas y publicaciones.

El artículo "Analysis of the Perceptual Quality Performance of Different HEVC Coding Tools" [10] ha proporcionado una validación perceptual rigurosa de las herramientas de codificación existentes en el estándar de vídeo HEVC. Más recientemente, en el artículo "A Hybrid Contrast and Texture Masking Model to Boost HEVC Perceptual RD Performance" [95], se propuso un modelo híbrido de enmascaramiento de contraste y textura que ha demostrado aumentar significativamente la eficiencia de codificación perceptual, logrando importantes ganancias en BD-Rate utilizando técnicas avanzadas de clasificación de bloques.

Además, numerosos trabajos presentados en congresos han permitido complementar el cuerpo de esta tesis, destacando el desarrollo del clasificador de bloques según su nivel de textura [14] y el análisis combinado del enmascaramiento por textura y contraste [11]. Estas contribuciones colectivas han reforzado el cumplimiento del objetivo principal de la tesis, aportando avances significativos en la codificación de vídeo y en la aplicación práctica de la percepción visual humana en la compresión de vídeo.

5.1. Contribuciones y resultados

5.1.1. Desarrollo de Software de Prototipado

El desarrollo del software de prototipado HEVC en Matlab ha sido una aportación fundamental de esta tesis. Este software nos ha permitido la simulación y validación de diversas técnicas de codificación perceptual, ofreciendo una herramienta flexible para la experimentación académica. A través de este emulador, se logró implementar y probar modelos de cuantificación perceptual, aplicando técnicas como el enmascaramiento por

contraste y textura. La generación de gráficas Rate-Distortion y métricas de rendimiento BD-Rate nos ha proporcionado una base cuantitativa robusta para evaluar el rendimiento perceptual del codificador, destacando su utilidad en las primeras etapas de esta investigación.

Publicación Clave:

- "Emulador HEVC INTRA en Matlab." J. R. Atencia, O. L. Granado, M. O. Martínez-Rach and M. P. Malumbres. Jornadas SARTECO, Cáceres, 18 a 20 de septiembre de 2019| / SARTECO (aut.), 2019, ISBN 9788409121274, págs. 351-359 [12].

5.1.2. Evaluación Perceptual de Herramientas de Codificación

Este estudio ha evaluado críticamente la respuesta perceptual de las herramientas de codificación incluidas en el estándar HEVC, utilizando métricas de calidad objetivas altamente correladas con la percepción humana, como como la SSIM, MS-SSIM y PSNR-HVS-M. Los análisis detallados revelaron que ciertas configuraciones específicas de estas herramientas ofrecen mejoras significativas en la percepción de la calidad visual, mientras se optimiza la tasa de bits. Esto ha demostrado la importancia de adoptar enfoques de evaluación más sofisticados que vayan más allá del tradicional PSNR.

Publicaciones Clave:

- "Analysis of the Perceptual Quality Performance of Different HEVC Coding Tools." J. R. Atencia, O. L. Granado, M. P. Malumbres, M. O. Martínez-Rach and G. Van Wallendael. **IEEE Access**. Vol. 9, pp. 37510-37522 (2021). DOI: 10.1109/ACCESS.2021.3062938 [10].

- "Análisis perceptual de las herramientas de codificación en el estándar HEVC." J. R. Atencia, O. L. Granado, M. O. Martínez-Rach, M. P. Malumbres and H. M. Gomis. Jornadas SARTECO, Málaga, 22 a 24 de septiembre de 2021| / SARTECO (aut.), 2021, ISBN-13 978-84-09-32487-3, págs. 289-302 [13].

5.1.3. Estudio e Integración de Técnicas de Codificación Perceptual

La investigación en técnicas de codificación perceptual, como el enmascaramiento por textura y contraste, ha permitido mejorar notablemente la eficiencia de compresión manteniendo una alta calidad visual. Durante la tesis, se implementaron y analizaron algoritmos específicos de enmascaramiento por textura y contraste, adaptándolos a las características del estándar HEVC y sus diversos tamaños de bloque. A través de un elaborado estudio se determinó una serie de matrices de pesos óptimas para cada tamaño de bloque, logrando una codificación eficiente bajo las condiciones perceptuales del sistema visual humano. La evaluación de diferentes técnicas de codificación perceptual combinadas reveló mejoras significativas en cuanto a calidad objetiva con métricas como la MS-SSIM y la PSNR-HVS-M, así como en la eficiencia de compresión medida por las métricas de distorsión BD-Rate.

El estudio demostró que la implementación de estas técnicas no solo mejora la percepción de calidad, sino que también ofrece un ahorro considerable en la tasa de bits. Estos hallazgos proporcionan una base sólida para la adopción de enfoques perceptuales en futuros estándares de codificación, asegurando que la calidad percibida se mantenga a pesar de las reducciones en la tasa de bits.

Publicaciones Clave:

- "A Hybrid Contrast and Texture Masking Model to Boost HEVC Perceptual RD Performance." J. R. Atencia, O. L. Granado, M. P. Malumbres, M. O. Martínez-Rach, D. Ruiz-Coll, G. Fernández-Escribano and G. Van Wallendael. **Electronics**, vol. 13, n. 16, 2024, doi: 10.3390/electronics13163341 [95].

- "Análisis combinado de texture y contrast masking en HEVC." J. R. Atencia, O. L. Granado, M. O. Martínez-Rach and M. P. Malumbres. Jornadas SARTECO, Cáceres, 18 a 20 de septiembre de 2019| / SARTECO (aut.), 2019, ISBN 9788409121274, págs. 301-308 [11].

5.1.4. Desarrollo de un Clasificador de Bloques

El desarrollo de un clasificador de bloques basado en algoritmos de aprendizaje supervisado ha permitido una clasificación efectiva de los bloques de imagen por nivel de textura. Esta herramienta ha facilitado la aplicación de técnicas de codificación perceptual ajustadas a las características específicas de cada bloque, mejorando así la eficiencia de compresión y la calidad visual de los vídeos codificados.

Publicaciones Clave:

- "A Hybrid Contrast and Texture Masking Model to Boost HEVC Perceptual RD Performance." J. R. Atencia, O. L. Granado, M. P. Malumbres, M. O. Martínez-Rach, D. Ruiz-Coll, G. Fernández-Escribano and G. Van Wallendael. **Electronics**, vol. 13, n. 16, 2024, doi: 10.3390/electronics13163341 [95].
- "Análisis de algoritmos de clasificación para la detección de texturas en bloques de codificación de vídeo." J. R. Atencia, O. L. Granado, M. O. Martínez-Rach, M. P. Malumbres and H. M. Gomis. Jornadas SARTECO, Ciudad Real, 20 a 22 de septiembre de 2023| / SARTECO (aut.), 2023, ISBN-13 978-84-09-54466-0, págs. 47-55 [14].

5.1.5. Integración en el Software de Referencia

La integración de técnicas avanzadas de enmascaramiento por textura y contraste en el software de referencia HEVC ha logrado reducciones significativas en la tasa de bits sin comprometer la calidad perceptual. Este logro subraya la efectividad de las técnicas desarrolladas y su aplicabilidad en entornos de codificación real, marcando un hito importante en la codificación de vídeo desde una perspectiva perceptual.

Publicaciones Clave:

- "A Hybrid Contrast and Texture Masking Model to Boost HEVC Perceptual RD Performance." J. R. Atencia, O. L. Granado, M. P. Malumbres, M. O. Martínez-Rach, D. Ruiz-Coll, G. Fernández-Escribano and G. Van Wallendael. **Electronics**, vol. 13, n. 16, 2024, doi: 10.3390/electronics13163341 [95].

- “Análisis combinado de texture y contrast masking en HEVC.” J. R. Atencia, O. L. Granado, M. O. Martínez-Rach and M. P. Malumbres. Jornadas SARTECO, Cáceres, 18 a 20 de septiembre de 2019| / SARTECO (aut.), 2019, ISBN 9788409121274, págs. 301-308 [11].

5.2. Desarrollos posteriores y futuros

A pesar de los avances significativos presentados en esta tesis, existen varias direcciones prometedoras para futuras investigaciones. La principal línea sería la adopción del estándar más reciente de codificación de vídeo, el Versatile Video Coding (VVC). En este caso, sería necesario estudiar en profundidad sus herramientas de codificación y evaluar su respuesta perceptual, así como integrar los modelos propuestos en esta tesis, con el objetivo de analizar si se obtienen las mismas ganancias.

Otro de los enfoques sería la exploración de nuevas técnicas de codificación que integren de manera más efectiva la inteligencia artificial para adaptarse dinámicamente a las características perceptuales del contenido de vídeo. El uso de redes neuronales convolucionales (CNN) podría ser útil en la clasificación de bloques, o incluso para extraer la información necesaria que permita determinar el nivel óptimo de sobre cuantificación aplicable.

Además, sería beneficioso explorar la implementación de estos modelos en sistemas de codificación en tiempo real, donde el valor del QP pueda adaptarse dinámicamente, no solo a las características espaciales del contenido de vídeo, sino también a las condiciones de transmisión. Esto permitiría mejorar la experiencia del usuario final, reduciendo el uso de ancho de banda sin sacrificar la calidad visual.

Finalmente, la evaluación de estas técnicas en diferentes resoluciones, desde contenido en 240p hasta 8K, podría proporcionar un mejor entendimiento sobre cómo ajustar las estrategias de optimización del QP según la resolución del contenido, explorando configuraciones de modelos duales que ofrezcan una mayor granularidad en la optimización para diferentes resoluciones.

BIBLIOGRAFÍA

[1] «2023 Global Internet Phenomena Report,» Sandvine, inf. téc., 2023. [En línea]. Disponible en: https://www.sandvine.com/global-internet-phenomena-report-2023.

[2] Cisco, «Cisco Annual Internet Report (2018–2023) White Paper,» Cisco, inf. téc., 2020. [En línea]. Disponible en: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html.

[3] Zhou Wang, A. C. Bovik, H. R. Sheikh y E. P. Simoncelli, «Image quality assessment: from error visibility to structural similarity,» *IEEE Transactions on Image Processing*, vol. 13, n.º 4, pp. 600-612, abr. de 2004. DOI: 10.1109/TIP.2003.819861.

[4] Z. Wang, E. P. Simoncelli y A. C. Bovik, «Multiscale structural similarity for image quality assessment,» en *The Thrity-Seventh Asilomar Conference on Signals, Systems Computers, 2003*, vol. 2, nov. de 2003, 1398-1402 Vol.2. DOI: 10.1109/ACSSC.2003.1292216.

[5] H. R. Sheikh y A. C. Bovik, «Image information and visual quality,» *IEEE Transactions on Image Processing*, vol. 15, n.º 2, pp. 430-444, feb. de 2006. DOI: 10.1109/TIP.2005.859378.

[6] K. O. Egiazarian, J. Astola, N. N. Ponomarenko, V. V. Lukin, F. Battisti y M. Carli, «A new full-reference Quality Metrics based on HVS,» 2006. [En línea]. Disponible en: https://api.semanticscholar.org/CorpusID:16871029.

[7] G. Bjontegaard, «Calculation of average PSNR differences between RD-Curves,» en *Proc. of the ITU-T Video Coding Experts Group - Thirteenth Meeting*, abr. de 2001.

[8] B. Girod, «What's Wrong with Mean-Squared Error?» En *Digital Images and Human Vision*. Cambridge, MA, USA: MIT Press, 1993, pp. 207-220.

[9] A. M. Eskicioglu y P. S. Fisher, «Image quality measures and their performance,» *IEEE Transactions on Communications*, vol. 43, n.º 12, pp. 2959-2965, 1995.

[10] J. R. Atencia, O. L. Granado, M. P. Malumbres, M. O. Martínez-Rach y GlennVan Wallendael, «Analysis of the Perceptual Quality Performance of Different HEVC Coding Tools,» *IEEE Access*, vol. 9, pp. 37 510-37 522, 2021. DOI: 10.1109/ACCESS.2021.3062938.

[11] J. R. Atencia, O. L. Granado, M. O. Martínez-Rach y M. P. Malumbres, «Análisis combinado de texture y contrast masking en HEVC,» en *Avances en Arquitectura y Tecnología de Computadores. Actas de las Jornadas SARTECO 2019*, ISBN: 9788409121274, 2019, pp. 301-308.

[12] J. R. Atencia, O. L. Granado, M. O. Martínez-Rach y M. P. Malumbres, «Emulador HEVC INTRA en Matlab,» en *Avances en Arquitectura y Tecnología de Computadores. Actas de las Jornadas SARTECO 2019*, ISBN: 9788409121274, 2019, pp. 351-359.

[13] J. R. Atencia, O. L. Granado, M. O. Martínez-Rach, M. P. Malumbres y H. M. Gomis, «Análisis perceptual de las herramientas de codificación en el estándar HEVC,» en *Avances en Arquitectura y Tecnología de Computadores. Actas de las Jornadas SARTECO 20/21*, ISBN: 9788409324873, 2021, pp. 289-302.

[14] J. R. Atencia, O. L. Granado, M. O. Martínez-Rach, M. P. Malumbres y H. M. Gomis, «Análisis de algoritmos de clasificación para la detección de texturas en bloques de codificación de vídeo,» en *Avances en Arquitectura y Tecnología de Computadores. Actas de las Jornadas SARTECO 2023*, ISBN: 9788409544660, 2023, pp. 47-55.

[15] P. Juluri, V. Tamarapalli y D. Medhi, «Measurement of Quality of Experience of Video-on-Demand Services: A Survey,» *IEEE Communications Surveys & Tutorials*, vol. 18, n.º 1, pp. 401-418, 2016. DOI: 10.1109/COMST.2015.2401424.

[16] H. R. Wu, A. R. Reibman, W. Lin, F. Pereira y S. S. Hemami, «Perceptual Visual Signal Compression and Transmission,» *Proceedings of the IEEE*, vol. 101, n.º 9, pp. 2025-2043, 2013. DOI: 10.1109/JPROC.2013.2262911.

[17] Z. Chen e Y. Li, «Recent advances in perceptual H.265/HEVC video coding,» en *2015 IEEE China Summit and International Conference on Signal and Information Processing (ChinaSIP)*, 2015, pp. 564-567. DOI: 10.1109/ChinaSIP.2015.7230466.

[18] H. Wei, X. Zhou, W. Zhou, C. Yan, Z. Duan y N. Shan, «Visual saliency based perceptual video coding in HEVC,» en *IEEE International Symposium on Circuits and Systems, ISCAS 2016, Montréal, QC, Canada, May 22-25, 2016*, IEEE, 2016, pp. 2547-2550. DOI: 10.1109/ISCAS.2016.7539112. [En línea]. Disponible en: http://dx.doi.org/10.1109/ISCAS.2016.7539112.

[19] J. Xu, Q. Peng, B. Wang, C. Li y X. Wu, «A Novel Saliency Based Bit Allocation and RDO for HEVC,» en *Advances in Multimedia Information Processing – PCM 2017*, B. Zeng, Q. Huang, A. El Saddik, H. Li, S. Jiang y X. Fan, eds., Cham: Springer International Publishing, 2018, pp. 70-78.

[20] S. Kim, D. Pak y S. Lee, «SSIM-based distortion metric for film grain noise in HEVC,» *Signal, Image and Video Processing*, vol. 12, mar. de 2018. DOI: 10.1007/s11760-017-1184-6.

[21] G. Xiang et al., «An improved adaptive quantization method based on perceptual CU early splitting for HEVC,» en *2017 IEEE International Conference on Consumer Electronics (ICCE)*, 2017, pp. 362-365. DOI: 10.1109/ICCE.2017.7889356.

[22] J. Ban, H. Lai y X. Lin, «A Novel Method Rate Distortion Optimization for HEVC Based on Improved SSIM,» en *2016 9th International Symposium on Computational Intelligence and Design (ISCID)*, vol. 2, 2016, pp. 260-263. DOI: 10.1109/ISCID.2016.2069.

[23] L. Prangnell y V. Sanchez, *Minimizing Compression Artifacts for High Resolutions with Adaptive Quantization Matrices for HEVC*, 2016. arXiv: 1609.06442 [cs.MM].

[24] M. S. Simon, A. Antony y G. Sreelekha, «Performance improvement in HEVC using contrast sensitivity function,» en *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2016, pp. 2237-2241. DOI: 10.1109/ICACCI.2016.7732384.

[25] L. Prangnell, «Visually lossless coding in HEVC: A high bit depth and 4:4:4 capable JND-based perceptual quantisation technique for HEVC,» *Signal Processing: Image Communication*, vol. 63, pp. 125-140, 2018. DOI: https://doi.org/10.1016/j.image.2018.02.007.

[26] H. Tong y A. Venetsanopoulos, «A perceptual model for JPEG applications based on block classification, texture masking, and luminance masking,» en *Proceedings 1998 International Conference on Image Processing. ICIP98 (Cat. No.98CB36269)*, oct. de 1998, 428-432 vol.3. DOI: 10.1109/ICIP.1998.999032.

[27] X. Zhang, W. Lin y P. Xue, «Improved estimation for just-noticeable visual distortion,» *Signal Processing*, vol. 85, n.º 4, pp. 795-808, 2005. DOI: 10.1016/j.sigpro.2004.12.002. [En línea]. Disponible en: http://www.sciencedirect.com/science/article/pii/S0165168404003263.

[28] D. Ruiz-Coll, G. Fernández-Escribano, J. L. Martínez y P. Cuenca, «Fast intra mode decision algorithm based on texture orientation detection in HEVC,» *Signal Processing: Image Communication*, vol. 44, pp. 12-28, mar. de 2016. DOI: https://doi.org/10.1016/j.image.2016.03.002.

[29] F. Wang, J. Chen, H. Zeng y C. Cai, «Spatial-frequency HEVC multiple description video coding with adaptive perceptual redundancy allocation,» *Journal of Visual Communication and Image Representation*, vol. 88, p. 103 614, 2022. DOI: https://doi.org/10.1016/j.jvcir.2022.103614. [En línea]. Disponible en: https://www.sciencedirect.com/science/article/pii/S1047320322001389.

[30] K. Yang, S. Wan, Y. Gong, H. R. Wu e Y. Feng, «Perceptual based SAO rate-distortion optimization method with a simplified JND model for H.265/HEVC,» *Signal Processing: Image Communication*, vol. 31, pp. 10-24, 2015. DOI: https://doi.org/10.1016/j.image.2014.11.005. [En línea]. Disponible en: https://www.sciencedirect.com/science/article/pii/S0923596514001635.

[31] P. Ndjiki-Nya, D. Bull y T. Wiegand, «Perception-oriented video coding based on texture analysis and synthesis,» en *2009 16th IEEE International Conference on Image Processing (ICIP)*, 2009, pp. 2273-2276. DOI: 10.1109/ICIP.2009.5414386.

[32] J. Kim, S.-H. Bae y M. Kim, «An HEVC-compliant perceptual video coding scheme based on JND models for variable block-sized transform kernels,» *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, n.º 11, pp. 1786-1800, 2015. DOI: 10.1109/TCSVT.2015.2389491.

[33] C. Hoffmann, S. Argyropoulos, A. Raake y P. Ndjiki-Nya, «Modelling image completion distortions in texture analysis-synthesis coding,» en *2013 Picture Coding Symposium (PCS)*, 2013, pp. 293-296. DOI: 10.1109/PCS.2013.6737741.

[34] V. Adzic, R. A. Cohen y A. Vetro, «Temporal perceptual coding using a visual acuity model,» en *Human Vision and Electronic Imaging XIX*, B. E. Rogowitz, T. N. Pappas y H. de Ridder, eds., International Society for Optics y Photonics, vol. 9014, SPIE, 2014, 90140P. DOI: 10.1117/12.2043130. [En línea]. Disponible en: https://doi.org/10.1117/12.2043130.

[35] S. Li, M. Xu, X. Deng y Z. Wang, «A novel weight-based URQ scheme for perceptual video coding of conversational video in HEVC,» en *2014 IEEE International Conference on Multimedia and Expo (ICME)*, 2014, pp. 1-6. DOI: 10.1109/ICME.2014.6890228.

[36] M. Xu, X. Deng, S. Li y Z. Wang, «Region-of-Interest Based Conversational HEVC Coding with Hierarchical Perception Model of Face,» *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, n.º 3, pp. 475-489, 2014. DOI: 10.1109/JSTSP.2014.2314864.

[37] L. Itti, «Automatic foveation for video compression using a neurobiological model of visual attention,» *IEEE Transactions on Image Processing*, vol. 13, n.º 10, pp. 1304-1318, 2004. DOI: 10.1109/TIP.2004.834657.

[38] Y. Li, W. Liao, J. Huang, D. He y Z. Chen, «Saliency based perceptual HEVC,» en *2014 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, 2014, pp. 1-5. DOI: 10.1109/ICMEW.2014.6890644.

[39] S. Milani, R. Bernardini y R. Rinaldo, «A saliency-based rate control for people detection in video,» en *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 2016-2020. DOI: 10.1109/ICASSP.2013.6638007.

[40] F. Liang, X. Peng y J.-Z. Xu, «Scene-aware perceptual video coding,» en *2013 Visual Communications and Image Processing (VCIP)*, nov. de 2013. [En línea]. Disponible en: https://www.microsoft.com/en-us/research/publication/scene-aware-perceptual-video-coding/.

[41] I. De y B. Chanda, «Multi-focus image fusion using a morphology-based focus measure in a quad-tree structure,» *Information Fusion*, vol. 14, n.º 2, pp. 136-146, 2013. DOI: https://doi.org/10.1016/j.inffus.2012.01.007. [En línea]. Disponible en: https://www.sciencedirect.com/science/article/pii/S1566253512000085.

[42] S. Pertuz, D. Puig y M. A. Garcia, «Analysis of focus measure operators for shape-from-focus,» *Pattern Recognition*, vol. 46, n.º 5, pp. 1415-1432, 2013. DOI: https://doi.org/10.1016/j.patcog.2012.11.011. [En línea]. Disponible en: https://www.sciencedirect.com/science/article/pii/S0031320312004736.

[43] H. Hadizadeh e I. V. Bajić, «Saliency-Aware Video Compression,» *IEEE Transactions on Image Processing*, vol. 23, n.º 1, pp. 19-33, 2014. DOI: 10.1109/TIP.2013.2282897.

[44] C. Yeo, H. L. Tan e Y. H. Tan, «SSIM-based adaptive quantization in HEVC,» en *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 1690-1694. DOI: 10.1109/ICASSP.2013.6637940.

[45] A. Rehman y Z. Wang, «SSIM-Inspired Perceptual Video Coding for HEVC,» en *2012 IEEE International Conference on Multimedia and Expo*, 2012, pp. 497-502. DOI: 10.1109/ICME.2012.175.

[46] ITU-T Recommendation, *H.261 : Video codec for audiovisual services at p x 64 kbit/s*, 1988.

[47] ISO/IEC 11172-2, *Information technology - Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s - Part 2: Video*, 1996.

[48] ITU-T Recommendation, *Information technology - Generic coding of moving pictures and associated audio information: Video*, 1996.

[49] ISO/IEC 13818-2, *Information technology - Generic coding of moving pictures and associated audio information - Part 2: Video*, 1996.

[50] ITU-T Recommendation, *Video coding for low bit rate communication*, 1998.

[51] ITU-T Recommendation, *Advanced video coding for generic audiovisual services*, 2003.

[52] ISO/IEC 14496-10, *Information technology - Coding of audio-visual objects - Part 10: Advanced video coding*, 2003.

[53] ITU-T Recommendation, *High efficiency video coding*, 2013.

[54] ISO/IEC 23008-2, *Information technology — High efficiency coding and media delivery in heterogeneous environments — Part 2: High efficiency video coding*, 2013.

[55] ITU-T Recommendation, *Versatile Video Coding*, 2020.

[56] ISO/IEC 23090-3:2021, *Information technology — Coded representation of immersive media — Part 3: Versatile video coding*, 2021.

[57] V. Sze, M. Budagavi y G. J. Sullivan, *High Efficiency Video Coding (HEVC): Algorithms and Architectures* (Integrated Circuits and Systems). Springer, 2014. DOI: 10.1007/978-3-319-06895-4.

[58] M. Budagavi, A. Fuldseth, G. Bjontegaard, V. Sze y M. Sadafale, «Core Transform Design for the High Efficiency Video Coding (HEVC) Standard,» *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, dic. de 2013. DOI: 10.1109/JSTSP.2013.2270429.

[59] A. Saxena y F. C. Fernandes, «DCT/DST-Based Transform Coding for Intra Prediction in Image/Video Coding,» *IEEE Transactions on Image Processing*, vol. 22, n.º 10, pp. 3974-3981, oct. de 2013. DOI: 10.1109/TIP.2013.2265882.

[60] J. D. Ruiz Coll, «Low Complexity HEVC Intra Coding,» Tesis doct., Universidad de Castilla-La Mancha, nov. de 2015. [En línea]. Disponible en: https://www.educacion.gob.es/teseo/mostrarRef.do?ref=1199766#.

[61] S. Daly, «Subroutine for the generation of a two dimensional human visual contrast sensitivity function,» *Eastman Kodak, Rochester, NY, Tech. Rep. Y*, vol. 233203, p. 1987, 1987.

[62] C. Fu et al., «Sample Adaptive Offset in the HEVC Standard,» *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, n.º 12, pp. 1755-1764, dic. de 2012. DOI: 10.1109/TCSVT.2012.2221529.

[63] M. O. Martínez-Rach, «Perceptual image coding for wavelet based encoders,» Tesis doct., Universidad Miguel Hernández de Elche, dic. de 2014. [En línea]. Disponible en: https://www.educacion.gob.es/teseo/mostrarRef.do?ref=1128660#.

[64] W. A. Shurcliff, «Studies in optics: A.A. Michelson, University of Chicago Press, 1927, republished in 1962 as Phoenix Science series no. 514 paperback, 176 pp. illustrated, $1.75,» *Journal of Physics and Chemistry of Solids*, vol. 24, pp. 498-499, 1963. [En línea]. Disponible en: https://api.semanticscholar.org/CorpusID:93011289.

[65] J. Mannos y D. Sakrison, «The effects of a visual fidelity criterion of the encoding of images,» *IEEE Transactions on Information Theory*, vol. 20, n.º 4, pp. 525-536, jul. de 1974. DOI: 10.1109/TIT.1974.1055250.

[66] K. N. Ngan, K. S. Leong y H. Singh, «Adaptive cosine transform coding of images in perceptual domain,» *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, n.º 11, pp. 1743-1750, nov. de 1989. DOI: 10.1109/29.46556.

[67] B. Chitprasert y K. R. Rao, «Human visual weighted progressive image transmission,» *IEEE Trans. on Communications*, vol. 38, n.º 7, pp. 1040-1044, jul. de 1990. DOI: 10.1109/26.57501.

[68] N. Nill, «A Visual Model Weighted Cosine Transform for Image Compression and Quality Assessment,» *IEEE Trans. on Communications*, vol. 33, n.º 6, pp. 551-557, jun. de 1985. DOI: 10.1109/TCOM.1985.1096337.

[69] Z. Wei y K. N. Ngan, «Spatio-Temporal Just Noticeable Distortion Profile for Grey Scale Image/Video in DCT Domain,» *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, n.º 3, pp. 337-346, mar. de 2009. DOI: 10.1109/TCSVT.2009.2013518.

[70] L. Ma, K. N. Ngan, F. Zhang y S. Li, «Adaptive Block-size Transform based Just-Noticeable Difference model for images/videos,» *Signal Processing: Image Communication*, vol. 26, n.º 3, pp. 162-174, 2011. DOI: https://doi.org/10.

1016/j.image.2011.02.002. [En línea]. Disponible en: http://www.sciencedirect.com/science/article/pii/S0923596511000191.

[71] J. Jin, D. Yu, W. Lin, L. Meng, H. Wang y H. Zhang, *Full RGB Just Noticeable Difference (JND) Modelling*, 2022. arXiv: 2203.00629 [eess.IV].

[72] Y. Wu, W. Ji y J. Wu, «Unsupervised Deep Learning for Just Noticeable Difference Estimation,» en *2020 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, 2020, pp. 1-6. DOI: 10.1109/ICMEW46912.2020.9105999.

[73] J. Park, J. Moon Jo y J. Jeong, «Some Adaptive Quantizers for HDTV Image Compression,» en dic. de 1994, pp. 417-423. DOI: 10.1016/B978-0-444-81844-7.50051-6.

[74] X. Zhang, W. Lin y P. Xue, «Just-noticeable Difference Estimation with Pixels in Images,» *J. Vis. Comun. Image Represent.*, vol. 19, n.º 1, pp. 30-41, ene. de 2008. DOI: 10.1016/j.jvcir.2007.06.001. [En línea]. Disponible en: http://dx.doi.org/10.1016/j.jvcir.2007.06.001.

[75] Fraunhofer Institute for Telecommunications. «HM Reference Software Version 16.20. » [En línea]. Disponible en: https://hevc.hhi.fraunhofer.de/trac/hevc/browser/tags/HM-16.20.

[76] Z. Li, A. Aaron, I. Katsavounidis, A. Moorthy y M. Manohara. «Toward a Practical Perceptual Video Quality Metric,» Netflix TechBlog. [En línea]. Disponible en: https://netflixtechblog.com/toward-a-practical-perceptual-video-quality-metric-653f208b9652.

[77] F. Bossen, D. Flynn, K. Sharman y K. Sühring, «HM Software Manual,» Joint Collaborative Team on Video Coding (JCTVC) of ITUT SG16 WP3 e ISO-IEC JTC1-SC29-WG11, inf. téc.

[78] A. Norkin et al., «HEVC Deblocking Filter,» *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, n.º 12, pp. 1746-1754, dic. de 2012. DOI: 10.1109/TCSVT.2012.2223053.

[79] G. J. Sullivan, J.-R. Ohm, W.-J. Han y T. Wiegand, «Overview of the High Efficiency Video Coding (HEVC) Standard,» *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, n.º 12, pp. 1649-1668, 2012. DOI: 10.1109/TCSVT.2012.2221191.

[80] J. Stankowski, C. Korzeniewski, M. Domański y T. Grajek, «Rate-distortion optimized quantization in HEVC: Performance limitations,» en *2015 Picture Coding Symposium (PCS)*, 2015, pp. 85-89.

[81] J. Kao, M. A. Hashemi, X. Xiu, Y. Ye, Y. He y J. Dong, «Improved transform skip mode for HEVC screen content coding,» en *2015 International Conference on Image Processing Theory, Tools and Applications (IPTA)*, nov. de 2015, pp. 504-509. DOI: 10.1109/IPTA.2015.7367197.

[82] C. Lan, J. Xu, G. J. Sullivan y F. Wu, «Intra transform skipping,» en *9th Meeting of the Joint Collaborative Team on Video Coding (JCT-VC)*, Document: JCTVC-I0408, abr. de 2012.

[83] G. Clare, F. Henri y J. Jung, «Sign Data Hiding,» en *7th Meeting of the Joint Collaborative Team on Video Coding (JCT-VC)*, Document: JCTVC-G271, nov. de 2011.

[84] F. Bossen, «Common test conditions and software reference,» en *11th Meeting of the Joint Collaborative Team on Video Coding (JCT-VC)*, Doc. JCTVC-K1100, oct. de 2012.

[85] H. R. Sheikh y A. C. Bovik, «A visual information fidelity approach to video quality assessmen,» en *The First International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, 2005, pp. 23-25.

[86] N. Ponomarenko, F. Silvestri, K. Egiazarian, M. Carli, J. Astola y V. Lukin, «On between-coefficient contrast masking of DCT basis functions,» en *Proceedings of the third international workshop on video processing and quality metrics*, vol. 4, 2007.

[87] R. Rassool, «VMAF reproducibility: Validating a perceptual practical video quality metric,» en *2017 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, jun. de 2017, pp. 1-2. DOI: 10.1109/BMSB.2017.7986143.

[88] J. G. L. K. P. L. C. Z. L. I. Katsavounidis, «VMA Fframework performance on UHD videos,» VQEG Meeting 2017 - Los Gatos, CA, USA, inf. téc., mayo de 2017.

[89] C. Lee, S. Woo, S. Baek, J. Han, J. Chae y J. Rim, «Comparison of objective quality models for adaptive bit-streaming services,» en *2017 8th International Conference on Information, Intelligence, Systems Applications (IISA)*, 2017, pp. 1-4. DOI: 10.1109/IISA.2017.8316385.

[90] D. Kundu y B. L. Evans, «Full-reference visual quality assessment for synthetic images: A subjective study,» en *2015 IEEE International Conference on Image Processing (ICIP)*, 2015, pp. 2374-2378. DOI: 10.1109/ICIP.2015.7351227.

[91] University of Southern California, Signal and Image Processing Institute. «The USC-SIPI Image Database. » [En línea]. Disponible en: http://sipi.usc.edu/database/database.php.

[92] N. Asuni y A. Giachetti, «TESTIMAGES: a Large-scale Archive for Testing Visual Devices and Basic Image Processing Algorithms,» en *Smart Tools and Apps for Graphics - Eurographics Italian Chapter Conference*, A. Giachetti, ed., The Eurographics Association, 2014. DOI: 10.2312/stag.20141242.

[93] Kodak. «The Kodak Color Image Dataset. » [En línea]. Disponible en: http://r0k.us/graphics/kodak/.

[94] Fraunhofer Institute for Telecommunications, *HM Reference Software Version 16.20*, https://hevc.hhi.fraunhofer.de/trac/hevc/browser/tags/HM-16.20, 2018.

[95] J. R. Atencia et al., «A Hybrid Contrast and Texture Masking Model to Boost High Efficiency Video Coding Perceptual Rate-Distortion Performance,» *Electronics*, vol. 13, n.º 16, 2024. DOI: 10.3390/electronics13163341. [En línea]. Disponible en: https://www.mdpi.com/2079-9292/13/16/3341.

Anexos

Anexo A:

Analysis of the Perceptual Quality Performance of Different HEVC Coding Tools

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 1, 2021, accepted February 22, 2021, date of publication March 1, 2021, date of current version March 11, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3062938

Analysis of the Perceptual Quality Performance of Different HEVC Coding Tools

JAVIER RUIZ ATENCIA[1], OTONIEL LÓPEZ GRANADO[1], MANUEL PÉREZ MALUMBRES[1], (Senior Member, IEEE), MIGUEL ONOFRE MARTÍNEZ-RACH[1], (Member, IEEE), AND GLENN VAN WALLENDAEL[2], (Member, IEEE)

[1]Department of Computer Engineering, Miguel Hernández University of Elche, 03202 Elche, Spain
[2]IDMedia Laboratory, Ghent University, 9000 Ghent, Belgium

Corresponding author: Javier Ruiz Atencia (javier.ruiza@umh.es)

This work was supported in part by the Spanish Ministry of Science, Innovation and Universities under Grant RTI2018-098156-B-C54, in part by FEDER funds (MINECO/FEDER/UE), and in part by the Ministry of Innovation, Universities, Science and Digital Society of the Valencian Government under Grant GV/2019/020.

ABSTRACT Each new video encoding standard includes encoding techniques that aim to improve the performance and quality of the previous standards. During the development of these techniques, PSNR was used as the main distortion metric. However, the PSNR metric does not consider the subjectivity of the human visual system, so that the performance of some coding tools is questionable from the perceptual point of view. To further explore this point, we have developed a detailed study about the perceptual sensibility of different HEVC video coding tools. In order to perform this study, we used some popular objective quality assessment metrics to measure the perceptual response of every single coding tool. The conclusion of this work will help to determine the set of HEVC coding tools that provides, in general, the best perceptual response.

INDEX TERMS HEVC, perceptual coding, transform skip, RDOQ, deblocking filter, SAO, CSF, perceptual metrics.

I. INTRODUCTION

High Efficiency Video Coding (HEVC) is the latest video coding standard in force developed by the Joint Collaborative Team on Video Coding (JCT-VC) of the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG) standardization organizations. During the development of the standard, a set of working draft specifications, including the accepted proposals, was published. In addition, the HEVC Test Model (HM) reference software was provided, so that the different coding techniques proposed could be tested.

In 2013, the first version of the standard was released. New versions of the standard and the reference software, including the multi-view extensions (MV-HEVC), the range extensions (RExt), and the scalability extensions (SHVC), have been launched since then.

The main goal of the HEVC standard was to reduce the bit rate by up to 50% while maintaining the same subjective quality as the previous H.264/AVC standard, without increasing the complexity of the encoder. To accomplish this goal, the HEVC standard incorporates numerous coding techniques that attempt to reduce the bit rate without increasing the distortion. Many of these techniques are based on the previous H.264/AVC standard, while other novel features, such as Quad-tree partitioning or the Sample Adaptive Offset (SAO) filter, were also included.

Some coding tools in the HEVC standard include approaches that deal with the non-linear behavior of the Human Visual System (HVS), in order to take into account the subjective quality perceived by humans during the encoding process. In particular, HEVC provides the SCaling List (SCL) coding tool, which applies a non-uniform quantization to the transformed coefficients, depending on the HVS contrast sensitivity associated to their frequencies. The main idea is that higher quantization can be applied to the areas of the scene for which the HVS is less sensitive, i.e., the Just Noticeable Distortion (JND) concept [1]. Some studies also include HEVC profiles to manage the luminance masking effect for High Dynamic Range (HDR) video sequences, as in [2], [3], where the authors apply a non-uniform quantization profile based on the Intensity Dependent Quantization [4]; it is

The associate editor coordinating the review of this manuscript and approving it for publication was Li Minn Ang.

adaptively applied to each frame based on a tone-mapping operator. An important bit rate reduction is obtained for the same quality that was measured with the specifically designed HDR-VPD-2 quality assessment metric [5] and also through subjective tests.

Although the coding techniques incorporated in the HEVC standard have proven to be capable of reducing the bit rate, it cannot be guaranteed that they are optimized from a perceptual point of view. During the development of the coding techniques, the Peak Signal-to-Noise Ratio (PSNR) metric was used to measure the distortion. PSNR, like Mean Square Error (MSE), provides a quality score based on the pixel differences between the original and reconstructed images. It is well known that these metrics do not accurately reflect the perceptual assessment of quality [6]–[9]. However, in the existing literature, there seems to be conflicting evidence about the accuracy of PSNR as a video quality metric. In [10], the authors proved that PSNR follows a monotonic relationship with subjective quality in the case of full frame rate encoding, when the video content and the video encoder are fixed.

So, in order to properly asses the perceptual (i.e., HVS-like) performance of HEVC coding tools, we need to employ quality assessment metrics that provide quality scores highly correlated with the quality perceived by humans. By doing this, we will ensure that the HEVC coding tools are always evaluated to maximize the perceptual performance of the overall encoder, avoiding, as much as possible, the deployment of cumbersome and time-consuming subjective tests.

In this study, we analyze the perceptual performance of the several coding tools of the HEVC Test Model (HM) software, which concerns the visual quality of a reconstructed video sequence. We have encoded the whole set of video sequences included in the HEVC common test conditions [11] with different configuration setups in order to analyze their perceptual response, trying to understand which encoder configuration maximizes the averaged perceptual quality of the reconstructed video sequences. So, on the one hand, we modify the encoder by changing its configuration, and on the other hand, we use multiple sequences (different content) to obtain this average. Therefore, under these conditions, PSNR should not be used as a reference metric to obtain perceptually based conclusions [12].

Each configuration setup will determine which coding tools are enabled, so we may analyze not only their individual contribution to the perceptual performance but also their contribution in combination with other coding tools. In order to measure the video quality, we use a set of well-known image objective quality assessment metrics, as well as the new Video Multi-method Assessment Fusion (VMAF) quality metric developed by Netflix [13].

The main contribution of this article is based on the R/D performance analysis of several HEVC coding tools, in order to properly assess their impact on the perceptual quality of the decoded video. Most of the available studies about HEVC coding tools in the literature only work with the PSNR, but very few of them are interested in perceptual behavior; usually, they focus only on a specific part of the encoder. We have exhaustively analyzed the impact of different coding tools on the perceptual quality of the reconstructed videos, showing results that differ from the results provided by PSNR. These results may be useful for future studies in order to configure the video encoder to maximize the perceptual R/D performance.

The rest of the article is structured as follows. An overview of the different HEVC coding tools under study is presented in Section II. In Section III, the methodology used in this study is explained. Section IV shows the experimental results, while in Section V, we provide a brief discussion of the obtained results. Finally, Section VI summarizes the conclusions of this study, and some future research lines are pointed out.

II. HEVC PARAMETERS

The HEVC standard includes many configuration parameters that are used to enable or disable coding tools that improve the reconstructed quality, reduce bit stream size, or simplify encoder complexity.

These parameters allow us to tune the coding structure, motion estimation, quantization, entropy coding, slice coding, deblocking filter, and rate control, among others [14]. Inside these main coding parameter blocks, the user can enable or disable the use of any parameter, as well as create a user-defined behavior for a given parameter. For example, in the deblocking filter parameter block, the user can enable or disable the loop filter and also define the use of the loop filter across the slice boundaries (subparameter: LFCrossSliceBoundaryFlag). However, some of these user-defined behaviors for some coding tools (subparameters) may affect the subjective quality. In this article, we will focus on the general behavior of the HEVC codec when enabling or disabling the main coding parameter inside each parameter block.

In this work, we have selected the following configuration parameters for evaluation, since they have a high impact on the visual quality of the decoded video sequence: *Scaling List, Deblocking Filter, SAO Filter, Rate-Distortion Optimized Quantization*, and *Transform Skip*.

A. SCALING LIST

The HVS is not able to detect all spatial frequencies with the same accuracy [15]. Numerous studies over the past few decades have characterized the Contrast Sensitivity Function (CSF) [16]–[18] as the response of our HVS to contrast variations, showing that the human eye is least sensitive to the highest and lowest frequencies.

This CSF is implemented in the quantification stage of the HEVC standard and can be modified by the Scaling List parameter, with three available options. By default, the encoder applies a constant quantizer step size for all transform coefficients (ScalingList = 0) that does not consider the subjectivity of the HVS. However, the HEVC standard

$$\begin{bmatrix} 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \end{bmatrix}$$

(a) 8×8 flat quantization matrix

$$\begin{bmatrix} 16 & 16 & 16 & 16 & 17 & 18 & 21 & 24 \\ 16 & 16 & 16 & 16 & 17 & 19 & 22 & 25 \\ 16 & 16 & 17 & 18 & 20 & 22 & 25 & 29 \\ 16 & 16 & 18 & 21 & 24 & 27 & 31 & 36 \\ 17 & 17 & 20 & 24 & 30 & 35 & 41 & 47 \\ 18 & 19 & 22 & 27 & 35 & 44 & 54 & 65 \\ 21 & 22 & 25 & 31 & 41 & 54 & 70 & 88 \\ 24 & 25 & 29 & 36 & 47 & 65 & 88 & 115 \end{bmatrix}$$

(b) 8×8 weight quantization matrix

FIGURE 1. Default 8×8 quantization matrices for (a) ScalingList = 0 and (b) ScalingList = 1.

(a) (b)

FIGURE 2. Example of using deblocking filter in BlowingBubbles frame, encoded at QP37: (a) DB disabled, (b) DB enabled.

includes pre-defined weighting matrices (ScalingList = 1) that incorporate an implementation of the CSF. These non-flat matrices (like the one shown at Figure 2-b) define an additional scaling of the quantizer step, which varies with the transformed coefficient position, i.e., the base function frequency [19].

The results of the study carried out by [20] showed that, on average, the use of the weight quantization matrices provides better subjective quality results.

B. DEBLOCKING FILTER

This filter reduces the effect of blocking artifacts that are inherent in the nature of the encoder. It is used after block reconstruction, but its implementation is done within the coding loop, i.e., the reconstructed and filtered blocks will be taken as reference for other blocks (in-loop filter).

Its implementation is similar to the one used in the H.264 standard [21], but it is somewhat more simplified. In HEVC, the decoder can adaptively choose between applying two levels of the deblocking filter (normal or strong) or not applying it, depending on the adjacent blocks and a certain threshold [22]. As an example, Figure 2 compares the use of the deblocking filter on a part of a decoded picture. As can be seen, the grid effect disappears when the filter is active; however, regions that should not be filtered out are also blurred.

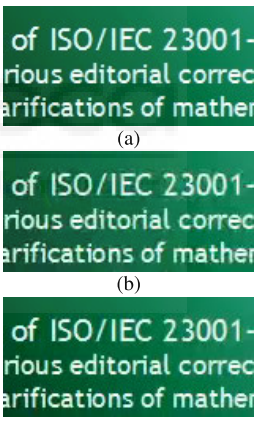
In [22], the authors argue that applying the deblocking filter increases the objective and subjective quality of the decoded video sequences.

C. SAO FILTER

The Sample Adaptive Offset (SAO) filter is a new algorithm integrated in the HEVC standard. It is located after the deblocking filter, and together, they form the so-called in-loop filter stage.

The main purpose of the SAO filter is to reduce distortion in the samples. To this end, the samples are classified into different categories, obtaining an offset for each of them. There are two sample processing techniques, band offset and edge offset. The algorithm will adaptively decide on the best strategy to use. The offset value is transmitted through the bit stream, while the classification of the samples is performed on both the encoder and decoder sides to reduce the information to be transmitted [23].

In [23], authors explain that using the SAO filter can provide about coding gains of 3.5% on average. To measure this gain, they have used the Bjøntegaard-Delta Rate (BD-Rate) metric [24], which uses the PSNR, a non-subjective metric. Regarding the subjective quality, the authors state that, based on experiments carried out by themselves, an improvement in quality is generally perceived. This improvement is higher in synthetic images, as shown in Figure 3, where SAO significantly improves the visual quality by suppressing the ringing artifacts near edges.



(a) (b) (c)

FIGURE 3. Example of using SAO filter in SlideEditing frame, encoded at QP32: (a) Original uncompressed (b) SAO disabled, (c) SAO enabled.

D. RATE-DISTORTION OPTIMIZED QUANTIZATION (RDOQ)

In the video encoder, optimizing the quantification process has a significant impact on the compression efficiency. The HEVC standard does not specify the quantization function, giving the encoder some flexibility in implementing it. HEVC includes, since version 13 of the reference software, a more sophisticated implementation of the quantization scheme called rate-distortion optimization quantization (RDOQ).

The purpose of RDOQ is to find the optimal or suboptimal set of quantized transform coefficients representing residual data in an encoded block. RDOQ calculates the image distortion (introduced by the quantization of transformed coefficients) in the encoded block and the number of bits needed to encode the corresponding quantized transform coefficients. Based on these two values, the encoder chooses, among

different coefficient blocks, the block which provides the better Rate Distortion (RD) cost [25].

Note that RDOQ is an effective method in terms of increasing the R/D performance. However, in [26], the authors claim that the PSNR-based mathematical reconstruction quality improvement attained by this technique is perceptually negligible in terms of how the human observer interprets the perceived quality of the compressed video data.

E. TRANSFORM SKIP

The Transform Skip parameter in the HEVC standard allows the encoder to bypass the transformation stage. In this way, the prediction errors are coded directly in the spatial domain.

During the development of HEVC, three transform skip modes were proposed and tested, but the standardization committee finally decided to use a single mode, the skipping transform in both the vertical and horizontal directions [27]. This mode was found to improve the compression of synthetic video sequences such as remote desktop, slideshows, etc.

Finally, the HM reference software includes an additional parameter, called TransformSkip-Fast, which enables or disables reduced testing of the transform-skipping mode decision in order to speed it up.

F. SIGN DATA HIDING

The transform coefficient coding in HEVC includes an option, called sign data hiding or sign bit hiding, that hides the coding of the sign flag of the first non-zero coefficient at the parity of the absolute sum of the coefficients. If the parity does not match the sign of the first non-zero coefficient and there are a sufficient number of significant coefficients, the encoder will modify the amplitude of a block coefficient until the desired sign is obtained [28].

14	3	7	**1**
9	-4	1	0
2	-4	0	0
-1	0	1	0

(a)

14	3	7	**0**
9	-4	1	0
2	-4	0	0
-1	0	1	0

(b)

FIGURE 4. Modified coefficient value example in 4 × 4 transform sub-block example for sign data hiding.

As an example, in Figure 4-a, we have a transformed sub-block of size 4 x 4, whose absolute sum is 47. By convention, an even value would derive a negative sign for the first non-zero coefficient. Following the block in zig-zag order from left to right and from top to bottom, the first non-zero coefficient is 14, which has a positive sign. This is why the encoder changes the value of a coefficient (Figure 4-b) so that the absolute sum will be even. The selection of the coefficient to be modified is determined by the Rate-Distortion criteria, which chooses the coefficient with the lowest R/D cost.

This compression technique achieves an average BD-Rate reduction of 0.6% for the All Intra coding mode. Modifying the coefficient values tends to increase the distortion, so the BD-Rate gain is obtained thanks to the rate reduction provided by this technique and not due to a quality increase.

III. METHODS AND PROCEDURES

In this work we have followed the indications established by the Common Test Condition [11]. This document defines a regulatory framework establishing a set of defined sequences and several base configurations for HM. The set of test sequences are classified in six large groups (A–F). The classes A, B, C, and D represent video sequences with different contents, video resolutions, frame rates, and bit depths. Class E is focused on head and shoulders videos typically used in video conference applications, and class F is devoted to computer generated videos and content screen applications (no natural video sequences).

In this work, we have focused only on the All Intra Main (AI Main) configuration mode, and therefore, no temporal processing and analysis was performed. We have only used the All Intra coding mode under the following criteria: (a) As most perceptual metrics are only available for images (not videos), the objective video quality measurement would be more accurate when using the All Intra coding mode since these metrics are unable to capture the motion-related artifacts. (b) Most of the coding tools analyzed in this article have a direct impact on the reconstruction quality as a result of a prediction process where the residual error is quantized and the entropy is encoded. So, with the independence of using spatial or temporal prediction, the quality distortion is due to the quantization of the prediction error. If two prediction blocks (one spatial and the other temporal) produce the same residual error, the reconstruction quality should be the same for a given quantization value, so the quality of the prediction and not the prediction itself (temporal or spatial) determines the final reconstruction quality.

Table 1 defines the set of test sequences used in this article.

In order to analyze the R/D performance of the coding tools described in the previous section, we use the Bjontegaard BD-Rate metric [24], which shows the rate-distortion performance. We followed the instructions defined in the HEVC conformance test standard [11]; the QPs 22, 27, 32, and 37 are used to conform the PSNR curves that allow the computation of the BD-Rate performance. As we are using other objective perceptual metrics like MS-SSIM and VMAF, we decided to add one more QP (QP = 42) in order to better fit the dynamic range response of these metrics, and as a consequence, provide more accurate BD-Rate results.

The BD-Rate values have been obtained for each metric and each coding configuration. These values show the bit rate savings (in percentage) between two rate-distortion curves. Due to the fact that the BD-Rate calculation was initially developed for the PSNR metric using third degree polynomial interpolation and four values per curve, the use of this algorithm applied to other metrics and with five points (QP values) per curve is not always optimal. Therefore, the

TABLE 1. HEVC test sequences.

Class	Sequence name	Resolution	Frame count	Frame rate	Bit depth
A	Traffic	2560x1600	150	30	8
	PeopleOnStreet		150	30	8
	Nebuta		300	60	10
	SteamLocomotive		300	60	10
B	Kimono	1920x1080	240	24	8
	ParkScene		240	24	8
	Cactus		500	50	8
	BQTerrace		600	60	8
	BasketballDrive		500	50	8
C	RaceHorses	832x480	300	30	8
	BQMall		600	60	8
	PartyScene		500	50	8
	BasketballDrill		500	50	8
D	RaceHorses	416x240	300	30	8
	BQSquare		600	60	8
	BlowingBubbles		500	50	8
	BasketballPass		500	50	8
E	FourPeople	1280x720	600	60	8
	Johnny		600	60	8
	KristenAndSara		600	60	8
F	BaskeballDrillText	832x480	500	50	8
	ChinaSpeed	1024x768	500	30	8
	SlideEditing	1280x720	300	30	8
	SlideShow	1280x720	500	20	8

interpolation method has been replaced by the Piecewise Cubic Hermite Interpolating Polynomial (PCHIP) [29] for higher accuracy.

To evaluate the influence of each parameter on the perceptual quality described in Section II, all test sequences have been coded by switching these parameters on and off, resulting in 64 configuration setups. These setups have been run using the reference software HEVC Test Model (HM) version 16.20 [30].

Due to the large number of measurements to be made, the use of subjective tests such as DMOS has been ruled out. Instead, we have proposed obtaining numerical values from Bjøntegaard-Delta rate measurements using the following objective quality metrics: SSIM, MS-SSIM, VIF, PSNR-HVS-M, and VMAF.

The SSIM (Structural Similarity) [9] and the MS-SSIM (Multi-Scale SSIM) [8] metrics are based on the hypothesis that the HVS is highly adapted to extract structural information from the scenes. Both metrics consider luminance, contrast, and structure information of the scenes, whereas MS-SSIM also considers the scale.

The VIF (Visual Information Fidelity) [31] metric uses the Natural Scene Statistics (NSS) model along with an image degradation model and components of the HVS to obtain the quality information.

The PSNR-HVS-M metric [32], a modified version of the PSNR, considers the contrast sensitivity function (CSF) and the between-coefficient contrast masking of DCT basis functions.

These metrics, unlike the PSNR, attempt to characterize the subjectivity of the HVS and do not include temporal information in their quality assessment algorithms.

The newest perceptual quality metric is the VMAF metric, developed by Netflix [13]. Unlike the previous metrics, VMAF makes use of novel machine learning techniques to estimate the result that would be obtained through subjective tests. To do that, this metric has been trained with inputs from real DMOS tests as well as three algorithms: VIF, DLM (Detail Loss Measure) [33], and TI (Temporal Perceptual Information) [34]. VIF measures the information fidelity loss, while DLM and TI measure the detail loss and the amount of motion, respectively.

Other works in the literature use VMAF: (a) in [35], the authors proved a strong correlation between subjective DMOS studies and the VMAF values obtained for a set of 4K sequences, (b) in [36], the authors also show a high correlation with the MOS values obtained for HD and UHD content, and (c) in [37], an analysis of different quality metrics for multi-resolution adaptive streaming showed that VMAF obtained the highest correlation with perceptual quality.

We have to say that the VMAF metric can be used for just one frame or for the whole video sequence. As the rest of the quality metrics only work at frame level, we decided to use VMAF for each frame in order to be coherent with the experiment setup and to avoid undesired effects when comparing all quality metrics results.

Regarding rate-distortion curves obtained in this work, the reference rate-distortion curve was obtained with the default All Intra Main configuration, whose parameters are shown in Table 2.

TABLE 2. Default values of the analyzed parameters.

Parameter	Value
QP	22, 27, 32, 37, 42
ScalingList	0
LoopFilterDisable	0
SAO	1
RDOQ	1
RDOQTS	1
TransformSkip	1
TransformSkipFast	1
SignHideFlag	1

IV. EXPERIMENTAL RESULTS

In this section, we show the results obtained after coding the set of test sequences when enabling and disabling the coding tools described in Section II with respect to the default configuration.

In order to measure the R/D performance of the different HEVC coding tool configurations, we have used the BD-Rate metric, as described in the previous section. The results from Tables 3(a) to 3(f) are provided by the BD-Rate metric, showing the R/D performance of the different coding tool setups measured by different perceptual quality metrics for all video sequences under evaluation (classes A to F). Negative values in these tables correspond to BD-Rate reductions or perceptual gains, and positive values correspond to BD-Rate increases or perceptual losses, with respect to the default or reference values (first row in Tables 3(a) to 3(f)).

TABLE 3. (a) Average coding performance [% BD-Rate] for Class A. (b) Average coding performance [% BD-Rate] for Class B. (c) Average coding performance [% BD-ate] for Class C. (d) Average coding performance [% BD-Rate] for Class D. (e) Average coding performance [% BD-Rate] for Class E. (f) Average coding performance [% BD-Rate] for Class F.

#	SCL	SAO	DB	RDOQ	TrSk	SSIM	MS-SSIM	VMAF	VIF	PSNR HVSM
1	**0**	**1**	**1**	**1**	**1**	0	0	0	0	0
2	0	1	1	1	0	−0.03	−0.03	0.06	−0.04	−0.04
3	0	1	1	0	1	5.6	2.29	11.93	3.84	5.38
4	0	1	1	0	0	5.54	2.24	11.95	3.8	5.34
5	0	1	0	1	1	0.26	1.03	1.32	−0.27	1.51
6	0	1	0	1	0	0.24	1	1.38	−0.31	1.46
7	0	1	0	0	1	5.91	3.46	13.48	3.61	7.05
8	0	1	0	0	0	5.86	3.42	13.51	3.57	7
9	0	0	1	1	1	0.18	0.21	−2.08	0.03	0.26
10	0	0	1	1	0	0.15	0.19	−2.03	−0.01	0.21
11	0	0	1	0	1	5.78	2.47	10.02	3.82	5.61
12	0	0	1	0	0	5.73	2.43	10.03	3.8	5.56
13	0	0	0	1	1	0.38	1.56	−3.32	−0.47	3.05
14	0	0	0	1	0	0.35	1.53	−3.27	−0.51	2.99
15	0	0	0	0	1	6.01	4.08	8.68	3.28	8.65
16	0	0	0	0	0	5.95	4.03	8.69	3.24	8.59
17	1	1	1	1	1	−0.38	−0.2	−0.18	−0.34	−0.63
18	1	1	1	1	0	−0.44	−0.24	−0.13	−0.38	−0.68
19	1	1	1	0	1	4.28	1.41	10.71	2.43	3.41
20	1	1	1	0	0	4.23	1.38	10.76	2.4	3.37
21	1	1	0	1	1	−0.11	0.81	1.12	−0.62	0.85
22	1	1	0	1	0	−0.17	0.77	1.18	−0.66	0.79
23	1	1	0	0	1	4.59	2.56	12.24	2.18	5.01
24	1	1	0	0	0	4.54	2.53	12.29	2.15	4.96
25	1	0	1	1	1	−0.19	0	−2.35	−0.34	−0.38
26	1	0	1	1	0	−0.25	−0.04	−2.29	−0.38	−0.43
27	1	0	1	0	1	4.46	1.6	8.78	2.42	3.63
28	1	0	1	0	0	4.4	1.57	8.86	2.39	3.59
29	1	0	0	1	1	0	1.34	−3.58	−0.85	2.37
30	1	0	0	1	0	−0.06	1.29	−3.53	−0.89	2.31
31	1	0	0	0	1	4.68	3.16	7.45	1.86	6.56
32	1	0	0	0	0	4.63	3.12	7.52	1.83	6.51

(a)

#	SCL	SAO	DB	RDOQ	TrSk	SSIM	MS-SSIM	VMAF	VIF	PSNR HVSM
1	**0**	**1**	**1**	**1**	**1**	0	0	0	0	0
2	0	1	1	1	0	−0.07	−0.06	−0.03	−0.06	−0.08
3	0	1	1	0	1	5.27	3.74	12.66	5.59	7.18
4	0	1	1	0	0	5.22	3.7	12.69	5.56	7.13
5	0	1	0	1	1	0.33	1.4	1.24	0.08	2.12
6	0	1	0	1	0	0.25	1.34	1.22	0.01	2.03
7	0	1	0	0	1	5.67	5.35	14.06	5.71	9.52
8	0	1	0	0	0	5.62	5.31	14.08	5.68	9.47
9	0	0	1	1	1	0.4	0.49	−1.92	−0.06	0.45
10	0	0	1	1	0	0.32	0.43	−1.95	−0.13	0.37
11	0	0	1	0	1	5.68	4.22	10.34	5.36	7.62
12	0	0	1	0	0	5.63	4.18	10.34	5.33	7.56
13	0	0	0	1	1	0.48	2.3	−2.94	−0.2	4.12
14	0	0	0	1	0	0.4	2.23	−2.98	−0.27	4.03
15	0	0	0	0	1	5.78	6.38	9.23	5.23	11.68
16	0	0	0	0	0	5.73	6.34	9.23	5.19	11.62
17	1	1	1	1	1	−0.72	−0.48	−0.47	−0.6	−1.06
18	1	1	1	1	0	−0.78	−0.52	−0.52	−0.71	−1.17
19	1	1	1	0	1	2.76	1.91	9.85	2.53	3.14
20	1	1	1	0	0	2.71	1.87	9.85	2.45	3.06
21	1	1	0	1	1	−0.4	0.9	0.75	−0.57	0.95
22	1	1	0	1	0	−0.45	0.85	0.71	−0.68	0.83
23	1	1	0	0	1	3.16	3.45	11.26	2.58	5.22
24	1	1	0	0	0	3.11	3.41	11.25	2.51	5.13
25	1	0	1	1	1	−0.34	−0.01	−2.47	−0.7	−0.63
26	1	0	1	1	0	−0.39	−0.05	−2.52	−0.82	−0.74
27	1	0	1	0	1	3.17	2.37	7.59	2.33	3.56
28	1	0	1	0	0	3.11	2.34	7.57	2.24	3.46
29	1	0	0	1	1	−0.26	1.77	−3.5	−0.89	2.92
30	1	0	0	1	0	−0.31	1.73	−3.55	−1.01	2.8
31	1	0	0	0	1	3.27	4.46	6.5	2.11	7.32
32	1	0	0	0	0	3.22	4.41	6.47	2.02	7.22

(b)

In these tables, we have omitted the SignHideFlag coding tool, as it does not provide significant changes in the image distortion (see Section IV-F). The SignHideFlag parameter is enabled in all tests, since this is its default value.

The complete set of tables, including the SignHideFlag parameter, for each of the classes, as well as for each specific video sequence, are available at the GATCOM research group's website [38].

To make the data in the tables easier to read, we have highlighted the cells with BD-Rate reductions as a heat map. The higher the reduction is, the greener the cell is. Each row corresponds to a specific permutation of the configuration parameters; the first row, highlighted in bold, shows the reference setup (default settings).

The first column is the permutation number, and it is used only as a reference in the text. The next five columns

TABLE 3. *(Continued.)* **(a) Average coding performance [% BD-Rate] for Class A. (b) Average coding performance [% BD-Rate] for Class B. (c) Average coding performance [% BD-ate] for Class C. (d) Average coding performance [% BD-Rate] for Class D. (e) Average coding performance [% BD-Rate] for Class E. (f) Average coding performance [% BD-Rate] for Class F.**

#	SCL	SAO	DB	RDOQ	TrSk	SSIM	MS-SSIM	VMAF	VIF	PSNR HVSM
1	**0**	**1**	**1**	**1**	**1**	0	0	0	0	0
2	0	1	1	1	0	−0.23	−0.22	−0.16	−0.18	−0.26
3	0	1	1	0	1	2.1	2.89	10.43	4.54	5
4	0	1	1	0	0	1.98	2.76	10.34	4.48	4.81
5	0	1	0	1	1	0.79	1	0.85	0.38	1.65
6	0	1	0	1	0	0.57	0.78	0.67	0.2	1.38
7	0	1	0	0	1	3.01	4.06	11.25	4.95	6.75
8	0	1	0	0	0	2.89	3.92	11.14	4.88	6.55
9	0	0	1	1	1	0.71	0.59	−1.84	0.25	0.65
10	0	0	1	1	0	0.5	0.38	−2.06	0.07	0.4
11	0	0	1	0	1	2.89	3.56	8.02	4.78	5.76
12	0	0	1	0	0	2.79	3.44	7.85	4.72	5.58
13	0	0	0	1	1	1.39	2.02	−2.71	0.72	3.77
14	0	0	0	1	0	1.17	1.8	−2.93	0.54	3.51
15	0	0	0	0	1	3.73	5.33	7.13	5.32	9.14
16	0	0	0	0	0	3.63	5.19	6.96	5.25	8.95
17	1	1	1	1	1	−0.02	−0.2	−0.32	−0.09	−0.22
18	1	1	1	1	0	−0.27	−0.44	−0.55	−0.34	−0.56
19	1	1	1	0	1	0.8	1.22	7.75	2.99	2.97
20	1	1	1	0	0	0.59	0.99	7.46	2.83	2.66
21	1	1	0	1	1	0.77	0.75	0.51	0.25	1.35
22	1	1	0	1	0	0.51	0.5	0.26	−0.01	0.99
23	1	1	0	0	1	1.68	2.28	8.59	3.34	4.59
24	1	1	0	0	0	1.48	2.05	8.31	3.17	4.27
25	1	0	1	1	1	0.67	0.36	−2.24	0.14	0.37
26	1	0	1	1	0	0.42	0.13	−2.53	−0.12	0.04
27	1	0	1	0	1	1.57	1.85	5.4	3.2	3.65
28	1	0	1	0	0	1.37	1.62	5.02	3.04	3.35
29	1	0	0	1	1	1.32	1.74	−3.12	0.56	3.44
30	1	0	0	1	0	1.08	1.48	−3.41	0.29	3.08
31	1	0	0	0	1	2.36	3.5	4.49	3.66	6.94
32	1	0	0	0	0	2.16	3.26	4.12	3.5	6.63

(c)

#	SCL	SAO	DB	RDOQ	TrSk	SSIM	MS-SSIM	VMAF	VIF	PSNR HVSM
1	**0**	**1**	**1**	**1**	**1**	0	0	0	0	0
2	0	1	1	1	0	−0.12	−0.21	−0.22	−0.2	−0.28
3	0	1	1	0	1	−0.42	0.97	10.54	4.34	4.77
4	0	1	1	0	0	−0.45	0.78	10.4	4.25	4.55
5	0	1	0	1	1	1.76	0.38	0.6	0.06	0.92
6	0	1	0	1	0	1.61	0.15	0.33	−0.15	0.63
7	0	1	0	0	1	1.56	1.48	11.05	4.44	5.8
8	0	1	0	0	0	1.51	1.27	10.91	4.35	5.55
9	0	0	1	1	1	0.41	0.1	−1.94	0.09	0.28
10	0	0	1	1	0	0.29	−0.12	−2.27	−0.1	0
11	0	0	1	0	1	0.04	1.12	8.25	4.45	5.14
12	0	0	1	0	0	0.04	0.94	7.97	4.36	4.9
13	0	0	0	1	1	3.19	0.71	−2.68	0.22	2.27
14	0	0	0	1	0	3.05	0.47	−3	0.02	1.97
15	0	0	0	0	1	3.25	1.99	7.49	4.63	7.32
16	0	0	0	0	0	3.25	1.8	7.22	4.54	7.08
17	1	1	1	1	1	0.87	0.01	−0.33	−0.03	−0.19
18	1	1	1	1	0	0.71	−0.27	−0.54	−0.28	−0.52
19	1	1	1	0	1	0.67	−0.12	8.07	2.95	2.97
20	1	1	1	0	0	0.67	−0.3	7.82	2.82	2.69
21	1	1	0	1	1	2.56	0.34	0.24	−0.01	0.64
22	1	1	0	1	0	2.39	0.06	0.01	−0.27	0.29
23	1	1	0	0	1	2.53	0.31	8.62	2.98	3.84
24	1	1	0	0	0	2.5	0.12	8.37	2.85	3.54
25	1	0	1	1	1	1.28	0.07	−2.38	0.04	0.06
26	1	0	1	1	0	1.13	−0.21	−2.7	−0.21	−0.27
27	1	0	1	0	1	1.1	−0.01	5.8	3.03	3.28
28	1	0	1	0	0	1.13	−0.17	5.43	2.91	3.01
29	1	0	0	1	1	4.02	0.63	−3.11	0.12	1.97
30	1	0	0	1	0	3.86	0.35	−3.43	−0.14	1.62
31	1	0	0	0	1	4.19	0.76	5.05	3.13	5.34
32	1	0	0	0	0	4.21	0.58	4.67	3.01	5.05

(d)

correspond to the enabling/disabling status of the following coding tools: SCL corresponds to ScalingList; SAO corresponds to SAO filter; DB corresponds to the inverse logic of the LoopFilterDisable parameter, that is, disabling DB means disabling the Deblocking filter; RDOQ includes both the RDOQ and RDOQTS coding tools; and TrSk includes both TransformSkip and TransformSkipFast.

The values scored by the metrics are not normalized, so each metric provides results in a different scale. However, we express the results in terms of the BD-Rate performance metric (percentage of rate reduction/increase), so we can compare results, hiding the real scale of each metric.

We have also performed a time profile of every single coding tool analyzed in this study to determine their average coding complexity. Considering both evaluation metrics, the perceptual R/D and the coding complexity, we may propose the proper coding tool configuration that better perceptual results provide with a balanced coding complexity.

TABLE 3. ***(Continued.)*** **(a) Average coding performance [% BD-Rate] for Class A. (b) Average coding performance [% BD-Rate] for Class B. (c) Average coding performance [% BD-ate] for Class C. (d) Average coding performance [% BD-Rate] for Class D. (e) Average coding performance [% BD-Rate] for Class E. (f) Average coding performance [% BD-Rate] for Class F.**

#	SCL	SAO	DB	RDOQ	TrSk	SSIM	MS-SSIM	VMAF	VIF	PSNR HVSM
1	**0**	**1**	**1**	**1**	**1**	0	0	0	0	0
2	0	1	1	1	0	−0.06	−0.06	−0.08	−0.07	−0.1
3	0	1	1	0	1	2.98	2.26	8.85	3.46	4.65
4	0	1	1	0	0	2.89	2.19	8.72	3.41	4.56
5	0	1	0	1	1	2.14	2.24	1.28	0.42	2.7
6	0	1	0	1	0	2.08	2.17	1.2	0.35	2.59
7	0	1	0	0	1	5.39	4.76	10.31	3.86	7.46
8	0	1	0	0	0	5.3	4.69	10.18	3.81	7.37
9	0	0	1	1	1	0.59	0.74	−2.14	0.28	0.68
10	0	0	1	1	0	0.53	0.67	−2.23	0.21	0.58
11	0	0	1	0	1	3.64	3.1	6.4	3.69	5.31
12	0	0	1	0	0	3.54	3.02	6.29	3.63	5.22
13	0	0	0	1	1	3.35	3.8	−2.61	0.75	4.96
14	0	0	0	1	0	3.29	3.73	−2.69	0.67	4.85
15	0	0	0	0	1	6.77	6.57	5.94	4.15	9.87
16	0	0	0	0	0	6.67	6.49	5.84	4.09	9.77
17	1	1	1	1	1	−0.59	−0.44	−0.25	−0.43	−0.77
18	1	1	1	1	0	−0.64	−0.49	−0.34	−0.5	−0.85
19	1	1	1	0	1	1.5	1.16	7.5	2.25	2.77
20	1	1	1	0	0	1.43	1.11	7.42	2.21	2.71
21	1	1	0	1	1	1.53	1.77	1	−0.03	1.86
22	1	1	0	1	0	1.48	1.72	0.91	−0.11	1.77
23	1	1	0	0	1	3.86	3.6	8.9	2.6	5.47
24	1	1	0	0	0	3.78	3.55	8.82	2.55	5.39
25	1	0	1	1	1	0	0.3	−2.53	−0.18	−0.11
26	1	0	1	1	0	−0.05	0.25	−2.63	−0.25	−0.2
27	1	0	1	0	1	2.16	1.99	4.95	2.46	3.41
28	1	0	1	0	0	2.09	1.95	4.89	2.42	3.34
29	1	0	0	1	1	2.73	3.33	−2.99	0.26	4.1
30	1	0	0	1	0	2.68	3.27	−3.09	0.18	4
31	1	0	0	0	1	5.22	5.38	4.49	2.86	7.82
32	1	0	0	0	0	5.14	5.32	4.42	2.81	7.74

(e)

#	SCL	SAO	DB	RDOQ	TrSk	SSIM	MS-SSIM	VMAF	VIF	PSNR HVSM
1	**0**	**1**	**1**	**1**	**1**	0	0	0	0	0
2	0	1	1	1	0	3.9	4.68	3.38	5.52	4.29
3	0	1	1	0	1	2.33	1.11	5.39	2.2	2.58
4	0	1	1	0	0	6.65	6.21	10.24	9.23	7.43
5	0	1	0	1	1	1.11	1.3	0.32	−0.48	1.55
6	0	1	0	1	0	5.13	6.11	3.73	5.8	5.9
7	0	1	0	0	1	3.53	2.56	5.74	2.46	4.17
8	0	1	0	0	0	7.91	7.74	10.6	9.5	9.09
9	0	0	1	1	1	1.36	1.56	0.7	0.95	1.72
10	0	0	1	1	0	5.43	6.59	3.15	7.95	5.8
11	0	0	1	0	1	3.7	2.77	6.43	3.93	4.29
12	0	0	1	0	0	8.08	8.06	10.04	11.67	8.86
13	0	0	0	1	1	2.56	3.19	0.27	1.26	4.07
14	0	0	0	1	0	6.77	8.38	2.75	8.29	8.31
15	0	0	0	0	1	5.02	4.61	5.99	4.25	6.76
16	0	0	0	0	0	9.52	10.06	9.64	12.02	11.51
17	1	1	1	1	1	−0.29	−0.11	−0.25	−0.82	−0.2
18	1	1	1	1	0	3.55	4.63	3.2	5.39	4.06
19	1	1	1	0	1	1.06	0.37	4.18	1.35	1.45
20	1	1	1	0	0	5.4	5.54	8.99	8.38	6.26
21	1	1	0	1	1	0.8	1.18	0.05	−0.58	1.31
22	1	1	0	1	0	4.76	6.03	3.55	5.64	5.63
23	1	1	0	0	1	2.25	1.8	4.54	1.59	2.98
24	1	1	0	0	0	6.66	7.05	9.45	8.64	7.86
25	1	0	1	1	1	1.04	1.46	0.42	0.84	1.48
26	1	0	1	1	0	5.07	6.55	2.89	7.77	5.54
27	1	0	1	0	1	2.39	1.96	5.2	3.03	3.09
28	1	0	1	0	0	6.82	7.44	8.81	10.76	7.61
29	1	0	0	1	1	2.24	3.05	0	1.12	3.78
30	1	0	0	1	0	6.41	8.3	2.49	8.08	8.01
31	1	0	0	0	1	3.68	3.77	4.77	3.32	5.5
32	1	0	0	0	0	8.24	9.42	8.42	11.09	10.21

(f)

In the following subsections, we will describe the results obtained showing the perceptual behavior of each coding tool under study, and in the next section, an analysis and discussion of these results will be provided.

A. SCALING LIST

When activating the ScalingList coding tool, a non-uniform quantization based on the contrast sensitivity function (CSF) is applied in the encoder quantization process. By default, it is disabled, so we have analyzed the perceptual influence of enabling it.

The results show that enabling the Scaling List parameter is perceptually beneficial for almost all settings and perceptual metrics. This can be seen by comparing rows 1 to 16 (SCL disabled) with rows 17 to 32 (SCL enabled) of the result tables, where the second group of coding settings

generally has a lower BD-Rate value than the first group. Taking into account the base configuration (row 1), just by enabling only the ScalingList coding tool, all perceptual metrics report BD-Rate reductions for all test video sequence classes. Regarding SCL coding complexity, when it is enabled the average encoding time increases between 3.47% and 8.44%, depending on the applied quantization (QP), as shown in Table 4.

From the results in Tables 3(a) to 3(f), we can extract the following main results: (a) when enabling the SCL coding tool (no matter the status of the rest of the coding tools), we obtain average BD-Rate reductions with all objective quality metrics and video classes (from 0.7% with SSIM to 1.4% with PSNR-HVS); (b) when combining the SCL and RDOQ coding tools, the BD-Rate saving increases an additional 0.9% on average for all objective video quality metrics, so both coding tools complement each other in terms of R/D performance; (c) an exception should be noticed with class D video sequences and the SSIM metric, where enabling the SCL coding tool shows an average BD-Rate increase of 0.9%. (d) The best result was provided by the VMAF metric, scoring an average BD-Rate reduction of 3.58% when both in-loop filters are disabled (SAO and DB) and RDOQ is enabled in class A video sequences.

B. DEBLOCKING FILTER

The deblocking filter minimizes the blocking effect produced by the block partitioning of images during the encoding process. By disabling this filter, the blocking artifacts become visible as the QP value increases.

TABLE 4. Average relative CPU encoding time increase/decrease [%], when enabling/disabling a single coding tool from the default encoder configuration (negative values mean time savings).

	QP 22	QP 27	QP 32	QP 37	QP 42	Avg.
SCL on	7.41	8.44	6.57	5.18	3.47	6.21
SAO off	−0.3	−0.21	−0.53	−0.37	−0.58	−0.4
DB off	−0.22	−0.18	−0.41	−0.24	−0.26	−0.26
RDOQ off	−16.56	−10.77	−6.4	−1.89	1.5	−6.82
TrSkp off	−15.78	−14.83	−14.24	−13.29	−13.22	−14.27
SBH off	−3.25	−2.72	−2.17	−1.49	−1.03	−2.13

As can be seen in Tables 3(a) to 3(f), better perceptual performance is provided when the DB coding tool is enabled, independently of the status of the rest of coding tools. This general behavior is observed in all video classes with all the objective quality metrics. The average BD-Rate improvement when enabling DB depends on every objective quality metric and also on the video class. For example, the SSIM, MS-SSIM, and PSNR-HVS metrics always report average BD-Rate savings of 1.2%, 1.5%, and 2.4%, respectively. However, VMAF (in all video classes) and VIF (in classes A and B) report BD-Rate savings when disabling the DB and SAO coding tools (between 0.4% and 1.1% BD-Rate savings). With respect to coding complexity, when DB filter is disabled, an average 0.3% reduction of the encoding time is observed, as shown in Table 4.

C. SAO FILTER

The SAO filter is a technique that attempts to minimize the distortion that is mainly introduced by the quantization step.

We find that in most cases, the perceptual metrics get higher BD-Rate values when disabling the SAO filter. This perceptual worsening is more significant in the synthetic video sequences (class F).

However, the VMAF metric gets better (lower) BD-Rate values for all video classes; it obtains even better results than the results obtained by the default configuration when the SAO filter is disabled, especially if the DB filter is also disabled. The BD-Rate reductions range from 1.8% to 4.4%; the best results occur when both in-loop filters are disabled. When working with class F videos, the VMAF BD-Rate reductions are very low when the SAO filter is disabled; they are always under 1%.

The VIF metric shows a similar behavior to VMAF when working with video classes A and B, achieving up to 0.5% BD-Rate savings when SAO is disabled (0.2% on average).

Finally, as with the deblocking filter, when disabling this filter no significant impact on coding complexity is observed, showing also an average encoding time reduction of 0.5%, as shown in Table 4.

D. RATE-DISTORTION OPTIMIZED QUANTIZATION (RDOQ)

The RDOQ algorithm achieves an estimated optimal quantization value that minimizes the Rate-Distortion cost. In this analysis, we have also disabled the RDOQTS parameter, which deactivates the RDOQ calculation for blocks marked as Transform Skip.

Although the PSNR metric is used by the RDOQ algorithm to measure distortion, looking at the results, we can see that disabling RDOQ parameters implies a deterioration of BD-Rate values for most of the perceptual metrics and video sequence classes. The benefits of enabling RDOQ are more remarkable for the VMAF metric, where on average, 9.8% BD-Rate savings are achieved by enabling the RDOQ coding tool.

When disabling the RDOQ tool, the coding complexity varies depending on the quantization parameter. The highest encoding time reductions are obtained when low quantization values are used (15.78% reduction at QP=22), as shown in Table 4. As the QP value increases, the encoding time savings are progressively reduced.

E. TRANSFORM SKIP

The TransformSkip and TransformSkipFast parameters are enabled by default, since they are able to obtain great BD-Rate savings for artificial or synthetic videos (those belonging to class F).

In Table 3(f), corresponding to synthetic or artificial video sequences, we can observe that disabling transform skip parameters causes a significant deterioration of BD-Rate values: all perceptual metrics get BD-Rate increases ranging from 3.6% to 7%.

If we analyze the other video classes, we can see that all of them have slight BD-Rate reductions when disabling the

transform skip parameters; these reductions are mostly close to 0% and are never higher than 0.4%.

When disabling the TransformSkip tool, the average encoding time is significantly affected. As stated in Section II-E, disabling this tool reduces the encoding time by almost 15%, as shown in Table 4.

F. SIGN DATA HIDING

The SignHideFlag coding tool, which is activated by default, enables a data compression technique called Sign Data Hiding that provides an average reduction of 0.6% in BD-Rate, regardless of the rest of the settings, as seen in Section II-F.

This coding tool produces a relatively high reduction in rate compared with the very low distortion that it produces, i.e., the perceived quality reported by the perceptual metrics remains almost the same, but the rate is reduced much more.

TABLE 5. Average coding performance [% BD-Rate] by disabling Sign Data Hiding to default configuration.

Class	SSIM	MS-SSIM	VMAF	VIF	PSNR HVSM
Class A	0.93	1.18	0.79	1.18	1.12
Class B	0.84	0.95	0.77	1.04	0.94
Class C	0.88	0.83	0.74	0.84	0.9
Class D	1.02	1	0.68	0.89	0.99
Class E	0.52	0.59	0.6	0.68	0.68
Class F	1.05	1.03	0.67	−0.12	1.08
Average	0.87	0.93	0.71	0.75	0.95

Table 5 shows the BD-Rate values obtained by disabling this parameter in the default configuration. As can be seen, by disabling this algorithm, an average BD-Rate increase from 0.71% to 0.95% is achieved.

Since this technique barely distorts the images, we have not included the column corresponding to this parameter in the results tables. Instead, it has been kept in its default state (enabled). The complete tables, including the results of the analysis of the SignDataHiding coding tool, will be available on the GATCOM research group website [38].

V. DISCUSSION

As shown in the previous section, there are coding tools that are perceptually ranked in the same way using all objective quality metrics in every video sequence class, and other coding tools have different behaviors depending on the video classes and/or the objective quality metric used. So, in this section, we will discuss and analyze in detail the results of the encoding tools, taking into account the relationships among them, the metrics and classes, and the reported perceptual behavior.

As stated previously, enabling the Transform Skip parameter works better in the sequences of class F, whereas for the rest of the classes, disabling it slightly increases the perceptual report given by almost all metrics. So, for classes A to E, we can simplify the analysis of the rest of coding tools by disabling Transform Skip and only enabling it when working with videos of class F.

When enabling the Scaling List (SCL) coding tool, we can see that a better perceptual response is reported by almost all metrics, achieving average BD-Rate savings of 0.9%. The behavior of the SCL coding tool is the one expected, since it is well known in the literature that the use of a CSF-based quantizer, implemented by the HEVC through the scaling list coding tool, improves the subjective quality of decoded video [20].

When disabling the RDOQ coding tool, all objective quality metrics show significative BD-Rate increases in all video classes. The average increment of BD-Rate provided in that case, for all metrics and video sequences, is about 4.16%, rising up to 11.5% for VMAF when working with class A. Therefore, we do not recommend deactivating the RDOQ parameter in any case. As there is for every general rule, there is an exception. In this case, the SSIM metric perceives an average BD-Rate reduction of 0.08% in sequences of class D; this goes up to 0.4% when disabling SCL and enabling both in-loop filters, again showing the inability of SSIM to properly score the class D video sequences when compared with the rest of the objective quality metrics.

From the previous analysis, we have shown that in order to provide a better perceptual quality performance for all objective metrics and video sequence classes, we have to (a) enable the SCL and RDOQ coding tools and (b) only enable the TrSk when working with class F video sequences (artificial or synthetic contents).

So, from now on, we will consider that the SCL and RDOQ coding tools are always enabled, and the TrSk is only enabled for class F video sequences. Under this assumption, we will analyze the behavior of the in-loop filters. In Table 6, we show the BD-Rate results of the SAO and DB configurations for each video class, keeping in mind that the rest of the coding tools are enabled/disabled as mentioned above. We have highlighted the maximum average BD-Rate savings of each quality metric and video class.

As can be seen, the general behavior of in-loop filters has two opposite positions: (a) the SSIM, MS-SSIM, and PSNR-VHS quality metrics provide the best perceptual results in all video classes when both filters are enabled, showing maximum BD-Rate savings of 0.28%, 0.35%, and 0.66%, respectively, and (b) VMAF (classes A to E) and VIF (classes A and B) say just the opposite, showing maximum BD-Rate savings of 3.40% and 0.95%, respectively, when both in-loop filters are disabled. However, it is worth saying that (a) when working with class F videos, there is a consensus between all metrics in enabling both in-loop filters to maximize the BD-Rate savings, (b) the VIF metric changes its scoring, suggesting that the best configuration for video classes C, D, and E is the one that enables both in-loop filters, joining the group formed by the SSIMM, MS-SSIM, and PSNR-VHS metrics, and (c) with respect to the VMAF metric, we have noticed that it is able to report average BD-Rate savings of 2.53% when only the DB filter is enabled and 0.45% when both filters are enabled.

Although all the objective quality metrics are designed to assess the quality in a way as close as possible to the way that the HVS does, each one uses a different approximation.

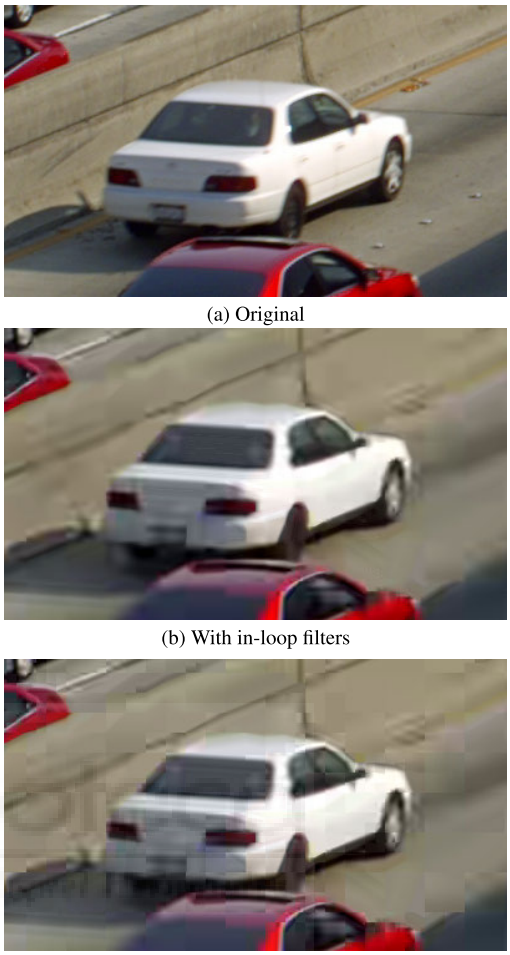
TABLE 6. BD-Rate evaluation of in-loop filters, with SCL=RDOQ=1 and TrSk=0 (=1 for class F).

SAO	DB	SSIM	MS-SSIM	VMAF	VIF	PSNR-HVSM
Class A						
1	1	**-0,44**	**-0,24**	-0,13	-0,38	**-0,68**
1	0	-0,17	0,77	1,18	-0,66	0,79
0	1	-0,25	-0,04	-2,29	-0,38	-0,43
0	0	-0,06	1,29	**-3,53**	**-0,89**	2,31
Class B						
1	1	**-0,78**	**-0,52**	-0,52	-0,71	**-1,17**
1	0	-0,45	0,85	0,71	-0,68	0,83
0	1	-0,39	-0,05	-2,52	-0,82	-0,74
0	0	-0,31	1,73	**-3,55**	**-1,01**	2,80
Class C						
1	1	**-0,27**	**-0,44**	-0,55	**-0,34**	**-0,56**
1	0	0,51	0,50	0,26	-0,01	0,99
0	1	0,42	0,13	-2,53	-0,12	0,04
0	0	1,08	1,48	**-3,41**	0,29	3,08
Class D						
1	1	**0,71**	**-0,27**	-0,54	**-0,28**	**-0,52**
1	0	2,39	0,06	0,01	-0,27	0,29
0	1	1,13	-0,21	-2,70	-0,21	-0,27
0	0	3,86	0,35	**-3,43**	-0,14	1,62
Class E						
1	1	**-0,64**	**-0,49**	-0,34	**-0,50**	**-0,85**
1	0	1,48	1,72	0,91	-0,11	1,77
0	1	-0,05	0,25	-2,63	-0,25	-0,20
0	0	2,68	3,27	**-3,09**	0,18	4,00
Class F						
1	1	**-0,29**	**-0,11**	**-0,25**	**-0,82**	**-0,20**
1	0	0,80	1,18	0,05	-0,58	1,31
0	1	1,04	1,46	0,42	0,84	1,48
0	0	2,24	3,05	0,00	1,12	3,78

Some of them perform the subband decomposition inspired by complex HVS models, while others extract structural information from the viewing field or even use the spatio-temporal statistical patterns found in signals captured from the visual field for which the HVS is adapted. Therefore, as we can see in this study, we obtain different quality assessments depending on the metric. In cases where all metrics report BD-Rate variations in the same direction, the conclusion is straightforward, but when the metrics' reports are opposite, a subjective test to validate the results is suggested.

In order to advance a preliminary subjective evaluation that sheds light on the metrics controversy around the in-loop filters' behavior, we have performed a simple subjective test with one class A video sequence. Class A has the highest BD-Rate differences (3.4%) found between the two options: enabling or disabling both in-loop filters (see Table 6). We have chosen a frame of a video sequence where the difference between the VMAF R/D curves of the two options is maximum. We have found that frame 22 of the Traffic_2560×1600_30 video sequence shows a 5.25% BD-Rate reduction when disabling both filters, taking as reference the configuration with the filters enabled. We have observed just noticeable perceptual differences at QPs 37 and 42 with respect to the original frame. These artifacts are more noticeable when the in-loop filters are disabled. In Figure 5, we show a cropped area of frame 22 (encoded with QP = 42), where the blocking artifacts are clearly visible when disabling both filters.

We have to mention that the results given by the VMAF are not biased by the image content or by the frame size,



(a) Original

(b) With in-loop filters

(c) Without in-loop filters

FIGURE 5. Comparison of cropped section of frame 22 of sequence Traffic (2560 × 1660) encoded with QP = 42, at 8.5*Mbps*.

as its results are consistent through different frame sizes and content. A good correlation with the DMOS and MOS values of the VMAF has been reported [35]–[37], showing that it can be considered a robust metric. Notice that although the behavior shown in Figure 5 seems to say that the VMAF metric does not correctly assess the perceived quality when both filters are disabled, it shows good results when only the DB filter is enabled. Although this observation does not mean that in-loop filters should always be enabled, a more detailed and carefully designed subjective evaluation test should be performed to determine the best in-loop filter configuration for the A to E video sequence classes.

Finally, another performance metric we may use to assess the most proper coding tool configuration is their contribution to the overall HEVC coding complexity. In Table 4 we have shown the time profiling results of each individual coding tool under study, showing their impact on the overall HEVC encoding complexity.

If we enable all coding tools but the TrSk (the best R/D perceptual configuration) the corresponding HEVC overall complexity will be reduced in an 14.27%, on average, when

compared to the default HEVC configuration. If we decide to also disable the in-loop filters, SAO and DB, we will get an additional coding time reduction of 0.66%.

VI. CONCLUSION

In this article, we have analyzed how the HEVC coding configuration parameters impact the perceptual rate-distortion. To do so, we have used the whole video sequence set defined in the HEVC common test conditions reference and obtained the Bjøntegaard Delta Rate (BD-Rate) measurements for a set of perceptual metrics widely used by the research community. Then, we analyzed how each HEVC coding tool impacts the perceptual BD-Rate and how this relates to other coding tools.

After analyzing the results provided by the set of HEVC coding tools under evaluation, we have arrived at the following conclusions:

a) The coding tools with the highest impact on the overall perceptual quality performance are RDOQ and SCL for all metrics reported in this study, so they should be always enabled.
b) TrSk should be enabled when working with class F videos (artificial, synthetic video contents), as significant perceptual gains are reported. However, for the rest of the video classes, it is slightly better to disable this coding tool. By disabling TrSk, the overall coding time is reduced by 15%.
c) The in-loop filters, SAO and DB, show opposite behaviors when working with video classes A to E, where
   i) one set of metrics (SSIM, MS-SSIM, and PSNR-HVS) implies that both filters should be enabled to maximize the perceptual BD-Rate savings,
   ii) VMAF implies that both filters should always be disabled, and
   iii) VIF shows the same results as VMAF for classes A and B, but for classes C, D, and E, it goes in the same direction as the other metrics.

The recommended HEVC coding tools configuration that will maximize the perceptual R/D should enable both SCL and RDOQ and disable TrSk (enabling it only with class F videos). As discussed in the previous section, there is no agreement with respect to the in-loop filters (SAO and DB). Three alternatives exist: (a) enable both filters, (b) disable them, and (c) only enable DB filter. The encoding complexity of both filters is low; therefore, their complexity does not help so much to take a firm decision. So, to determine the best option, we need to design specific subjective tests, taking into account the target video classes, to decide which one should be used.

The data presented in this article is intended to help other researchers to determine the best encoder configuration, depending on the type of sequence to be coded, to maximize the perceptual rate-distortion performance, taking into account the coding tools complexity. Also, it can be useful to choose the most appropriate perceptual metric to be used in the design of subjective tests.

As future work, we plan to extend this study by including more HEVC coding tools and perform exhaustive subjective tests to determine the perceptual-based settings that should be configured in the HEVC encoder to maximize the R/D quality performance.

REFERENCES

[1] H. R. Wu and K. R. Rao, *Digital Video Image Quality and Perceptual Coding* (Signal Processing and Communications). Boca Raton, FL, USA: CRC Press, 2005.
[2] Y. Zhang, M. Naccari, D. Agrafiotis, M. Mrak, and D. R. Bull, "High dynamic range video compression by intensity dependent spatial quantization in HEVC," in *Proc. Picture Coding Symp. (PCS)*, Dec. 2013, pp. 353–356.
[3] Y. Zhang, M. Naccari, D. Agrafiotis, M. Mrak, and D. R. Bull, "High dynamic range video compression exploiting luminance masking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 5, pp. 950–964, May 2016.
[4] M. Naccari, M. Mrak, D. Flynn, and A. Gabriellini, *Improving HEVC Compression Efficiency by Intensity Dependant Spatial Quantisation*, document JCTVC-J0076, 10th Meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Jul. 2012.
[5] P. Ren, J. Wang, J. Snyder, X. Tong, and B. Guo, "Pocket reflectometry," *ACM Trans. Graph.*, vol. 30, no. 4, pp. 1–10, Jul. 2011, doi: 10.1145/2010324.1964940.
[6] B. Girod, *What's Wrong With Mean-Squared Error?* Cambridge, MA, USA: MIT Press, 1993, pp. 207–220.
[7] A. M. Eskicioglu and P. S. Fisher, "Image quality measures and their performance," *IEEE Trans. Commun.*, vol. 43, no. 12, pp. 2959–2965, Dec. 1995.
[8] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. 37th Asilomar Conf. Signals, Syst. Comput.*, vol. 2, Nov. 2003, pp. 1398–1402.
[9] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[10] Q. Huynh-Thu and M. Ghanbari, "The accuracy of PSNR in predicting video quality for different video scenes and frame rates," *Telecommun. Syst.*, vol. 49, no. 1, pp. 35–48, Jan. 2012.
[11] F. Bossen, *Common Test Conditions and Software Reference* document JCTVC-K1100, 11th Meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Oct. 2012.
[12] Q. Huynh-Thu and M. Ghanbari, "Scope of validity of PSNR in image/video quality assessment," *Electron. Lett.*, vol. 44, no. 13, pp. 800–801, Jun. 2008.
[13] Z. Li, A. Aaron, I. Katsavounidis, A. Moorthy, and M. Manohara. (Jun. 2016). *Toward a Practical Perceptual Video Quality Metric*. Netflix TechBlog. [Online]. Available:https://netflixtechblog.com/toward-a-practical-perceptual-video-quality-metric-653f208b9652
[14] F. Bossen, D. Flynn, K. Sharman, and K. Sühring, *HM Software Manual*, document Joint Collaborative Team on Video Coding (JCT–VC) of ITU–T SG16 WP3 and ISO-IEC JTC1-SC29-WG11, 2020.
[15] M. O. Martínez-Rach, "Perceptual image coding for wavelet based encoders," Ph.D. dissertation, Dept. Syst. Eng. Automat., Miguel Hernández Univ. Elche, Alicante, Spain, Dec. 2014. [Online]. Available: https://www.educacion.gob.es/teseo/mostrarRef.do?ref=1128660#
[16] J. Mannos and D. Sakrison, "The effects of a visual fidelity criterion of the encoding of images," *IEEE Trans. Inf. Theory*, vol. IT-20, no. 4, pp. 525–536, Jul. 1974.
[17] N. Nill, "A visual model weighted cosine transform for image compression and quality assessment," *IEEE Trans. Commun.*, vol. COM-33, no. 6, pp. 551–557, Jun. 1985.
[18] B. Chitprasert and K. R. Rao, "Human visual weighted progressive image transmission," *IEEE Trans. Commun.*, vol. 38, no. 7, pp. 1040–1044, Jul. 1990.
[19] M. Wien, *High Efficiency Video Coding: Coding Tools and Specifications* (Signals and Communication Technology). Berlin, Germany: Springer, 2015.
[20] M. Haque, A. Tabatabai, and Y. Morigami, *HVS Model Based Default Quantization Matrices*, document JCTVC-G880, 7th Meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Nov. 2011.

[21] P. List, A. Joch, J. Lainema, G. Bjontegaard, and M. Karczewicz, "Adaptive deblocking filter," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 614–619, Jul. 2003.
[22] A. Norkin, G. Bjontegaard, A. Fuldseth, M. Narroschke, M. Ikeda, K. Andersson, M. Zhou, and G. Van der Auwera, "HEVC deblocking filter," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1746–1754, Dec. 2012.
[23] C.-M. Fu, E. Alshina, A. Alshin, Y.-W. Huang, C.-Y. Chen, C.-Y. Tsai, C.-W. Hsu, S.-M. Lei, J.-H. Park, and W.-J. Han, "Sample adaptive offset in the HEVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1755–1764, Dec. 2012.
[24] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," in *Proc. ITU-T Video Coding Experts Group-13th Meeting*, Apr. 2001, pp. 1–4.
[25] J. Stankowski, C. Korzeniewski, M. Domanski, and T. Grajek, "Rate-distortion optimized quantization in HEVC: Performance limitations," in *Proc. Picture Coding Symp. (PCS)*, May 2015, pp. 85–89.
[26] L. Prangnell, "Visually lossless coding in HEVC: A high bit depth and 4:4:4 capable JND-based perceptual quantisation technique for HEVC," *Signal Process., Image Commun.*, vol. 63, pp. 125–140, Apr. 2018.
[27] J.-Y. Kao, M. A. Hashemi, X. Xiu, Y. Ye, Y. He, and J. Dong, "Improved transform skip mode for HEVC screen content coding," in *Proc. Int. Conf. Image Process. Theory, Tools Appl. (IPTA)*, Nov. 2015, pp. 504–509.
[28] G. Clare, F. Henri, and J. Jung, *Sign Data Hiding*, document JCTVC-G271, 7th Meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Nov. 2011.
[29] F. N. Fritsch and R. E. Carlson, "Monotone piecewise cubic interpolation," *SIAM J. Numer. Anal.*, vol. 17, no. 2, pp. 238–246, Apr. 1980, doi: 10.1137/0717021.
[30] Fraunhofer Institute for Telecommunications. (Sep. 2018). *HM Reference Software Version 16.20*. [Online]. Available: https://hevc.hhi.fraunhofer.de/trac/hevc/browser/tags/HM-16.20
[31] H. R. Sheikh and A. C. Bovik, "A visual information fidelity approach to video quality assessmen," in *Proc. 1st Int. Workshop Video Process. Qual. Metrics Consum. Electron.*, 2005, pp. 23–25.
[32] N. Ponomarenko, F. Silvestri, K. Egiazarian, M. Carli, J. Astola, and V. Lukin, "On between-coefficient contrast masking of DCT basis functions," in *Proc. 3rd Int. Workshop Video Process. Qual. Metrics*, vol. 4, 2007, pp. 1–4.
[33] S. Li, F. Zhang, L. Ma, and K. N. Ngan, "Image quality assessment by separately evaluating detail losses and additive impairments," *IEEE Trans. Multimedia*, vol. 13, no. 5, pp. 935–949, Oct. 2011.
[34] *Subjective Video Quality Assessment Methods for Multimedia Applications*, Standard Recommendation ITU-T P.910, International Telecommunication Union, Apr 2008
[35] R. Rassool, "VMAF reproducibility: Validating a perceptual practical video quality metric," in *Proc. IEEE Int. Symp. Broadband Multimedia Syst. Broadcast. (BMSB)*, Jun. 2017, pp. 1–2.
[36] J. Gutiérrez, L. Krasula, P. L. Callet, Z. Li, and I. Katsavounidis, "VMAF framework performance on UHD videos," in *Proc. VQEG Meeting*, Los Gatos, CA, USA, May 2017, pp. 1–9.
[37] C. Lee, S. Woo, S. Baek, J. Han, J. Chae, and J. Rim, "Comparison of objective quality models for adaptive bit-streaming services," in *Proc. 8th Int. Conf. Inf., Intell., Syst. Appl. (IISA)*, Aug. 2017, pp. 1–4.
[38] Universidad Miguel Hernández de Elche. *Grupo de Arquitectura y Tecnología de Computadores*. Accessed: Feb. 22, 2021. [Online]. Available: http://atc.umh.es/gatcom/Ficheros/overall_results.zip and http://atc.umh.es/gatcom/

JAVIER RUIZ ATENCIA received the B.S. degree (Hons.) in telecommunications technology engineering and the M.S. degree in telecommunications engineering from the Miguel Hernández University of Elche, Spain, in 2014 and 2019, respectively, where he is currently pursuing the Ph.D. degree with the Department of Computer Engineering. His B.S. degree's last year was supported by the SICUE mobility program with the Polytechnic University of Catalonia. In 2012, he participated in the creation of the IEEE UMH Student Branch. From 2014 to 2019, he was with the Department of Research and Development with an important energy efficiency company. His research includes perceptual compression techniques applied to the latest image and video standards.

OTONIEL LÓPEZ GRANADO received the M.S. degree in computer science from the University of Alicante, Spain, in 1996, and the Ph.D. degree in computer science with the Miguel Hernández University of Elche, Spain, in 2010. From 1997 to 2003, he was a Programmer Analyst with an important industrial informatics firm. In 2003, he joined the Department of Computer Engineering, Miguel Hernández University of Elche, as an Assistant Professor, where he was promoted to an Associate Professor in 2012, where he currently leads the GATCOM Research Group. His research and teaching activities are related to multimedia networking, including audio/video coding and network delivery.

MANUEL PÉREZ MALUMBRES (Senior Member, IEEE) received the B.S. degree in computer science from the University of Oviedo, Spain, in 1986, and the M.S. and Ph.D. degrees in computer science from the Technical University of Valencia in 1991 and 1996, respectively. He has authored more than 200 conference and journal publications and several networking books for undergraduate CS courses. His current research and teaching activities are related to multimedia networking, including image/video coding and network delivery, wireless network technologies, including MANETs, VANETs, and WSNs, and acceleration schemas for multimedia applications, including multi/many-threads, GPUs, and FPGAs.

MIGUEL ONOFRE MARTÍNEZ-RACH (Member, IEEE) received the M.S. degree in computer science from the University of Alicante in 1996 and the Ph.D. degree (Hons.) from the Miguel Hernández University of Elche in 2014. He was with a multinational French-owned computer company as a Data Warehouse Analyst. In 2003, he joined the Miguel Hernández University of Elche, where he is currently an Associate Professor with the Department of Computer Engineering. His teaching subjects include operating systems and mobile programing. His research subjects are related to image and video compression, specifically with perceptual coding. His current research is focused on the perceptual enhancement of the HEVC standard.

GLENN VAN WALLENDAEL (Member, IEEE) received the M.Sc. degree in applied engineering from the University College of Antwerp, Belgium, in 2006, and the M.Sc. degree in engineering from Ghent University, Belgium, in 2008. He is currently pursuing the Ph.D. degree with the IDLab, Ghent University—IMEC. His Ph.D. degree is supported by the Research Foundation—Flanders (FWO), where he is currently working as a Post-Doctoral Researcher. His main topic of interest is video compression, including scalable video compression and transcoding.

Anexo B:

A Hybrid Contrast and Texture Masking Model to Boost High Efficiency Video Coding Perceptual Rate-Distortion Performance

electronics

IMPACT FACTOR 2.6

CITESCORE 5.3

Article

A Hybrid Contrast and Texture Masking Model to Boost High Efficiency Video Coding Perceptual Rate-Distortion Performance

Javier Ruiz Atencia, Otoniel López-Granado, Manuel Pérez Malumbres, Miguel Martínez-Rach, Damian Ruiz Coll, Gerardo Fernández Escribano and Glenn Van Wallendael

Special Issue

Recent Advances in Image/Video Compression and Coding

Edited by

Dr. Miroslav Uhrina, Dr. Jaroslav Frnda and Dr. Lukas Sevcik

MDPI

https://doi.org/10.3390/electronics13163341

electronics

MDPI

Article

A Hybrid Contrast and Texture Masking Model to Boost High Efficiency Video Coding Perceptual Rate-Distortion Performance

Javier Ruiz Atencia [1,*], Otoniel López-Granado [1,*], Manuel Pérez Malumbres [1], Miguel Martínez-Rach [1], Damian Ruiz Coll [2], Gerardo Fernández Escribano [3] and Glenn Van Wallendael [4]

[1] Department Computer Engineering, Miguel Hernández University, 03202 Elche, Spain; mels@umh.es (M.P.M.); mmrach@umh.es (M.M.-R.)
[2] Department of Signal and Communications Theory, Rey Juan Carlos University, 28933 Madrid, Spain; druizcoll@ofinno.com
[3] School of Industrial Engineering, University of Castilla-La Mancha, 13001 Albacete, Spain; gerardo.fernandez@uclm.es
[4] IDLab-MEDIA, Ghent University—IMEC, B-9052 Ghent, Belgium; glenn.vanwallendael@ugent.be
* Correspondence: javier.ruiza@umh.es (J.R.A.); otoniel@umh.es (O.L.-G.); Tel.: +34-96665-8392 (O.L.-G.)

Abstract: As most of the videos are destined for human perception, many techniques have been designed to improve video coding based on how the human visual system perceives video quality. In this paper, we propose the use of two perceptual coding techniques, namely contrast masking and texture masking, jointly operating under the High Efficiency Video Coding (HEVC) standard. These techniques aim to improve the subjective quality of the reconstructed video at the same bit rate. For contrast masking, we propose the use of a dedicated weighting matrix for each block size (from 4×4 up to 32×32), unlike the HEVC standard, which only defines an 8×8 weighting matrix which it is upscaled to build the 16×16 and 32×32 weighting matrices (a 4×4 weighting matrix is not supported). Our approach achieves average Bjøntegaard Delta-Rate (BD-rate) gains of between 2.5% and 4.48%, depending on the perceptual metric and coding mode used. On the other hand, we propose a novel texture masking scheme based on the classification of each coding unit to provide an over-quantization depending on the coding unit texture level. Thus, for each coding unit, its mean directional variance features are computed to feed a support vector machine model that properly predicts the texture type (plane, edge, or texture). According to this classification, the block's energy, the type of coding unit, and its size, an over-quantization value is computed as a QP offset (DQP) to be applied to this coding unit. By applying both techniques in the HEVC reference software, an overall average of 5.79% BD-rate gain is achieved proving their complementarity.

Keywords: HEVC; perceptual coding; HVS; CSF; texture masking; contrast masking; MDV; SVM; adaptive QP

Citation: Atencia, J.R.; López-Granado, O.; Pérez Malumbres, M.; Martínez-Rach, M.; Coll, D.R.; Fernández Escribano, G.; Van Wallendael, G. A Hybrid Contrast and Texture Masking Model to Boost High Efficiency Video Coding Perceptual Rate-Distortion Performance. *Electronics* **2024**, *13*, 3341. https://doi.org/10.3390/electronics13163341

Academic Editor: Krzysztof Okarma

Received: 27 May 2024
Revised: 6 August 2024
Accepted: 17 August 2024
Published: 22 August 2024

1. Introduction

Image and video compression standards play an essential role in modern media communication, enabling the efficient storage and transmission of digital content. However, the compression process unavoidably introduces some degree of information loss, resulting in image or video distortion that can be perceived by human observers. To improve the subjective quality of compressed media, many techniques based on the perception of the human visual system (HVS) have been developed.

The quantization stage is a crucial step in the image and video coding chain, where information is discarded to reduce the quantity of data to be stored or transmitted. This process introduces artifacts and distortions that are not present in the original source. Therefore, it is crucial to consider the limitations and properties of the HVS to develop efficient compression algorithms.

The masking properties of the HVS have been extensively studied to provide mechanisms to quantize the information of image areas where reconstruction errors are not perceived by the HVS [1]. The HVS is not always able to detect distortions when they are masked by texture, contrast, luminance, and other factors. Therefore, these masking properties can be used to reduce the perceptual impact of compression artifacts.

Contrast masking is one of the most commonly used HVS-based techniques to reduce compression artifacts. It involves incorporating the contrast sensitivity function (CSF) during the quantization stage of image and video codecs. The CSF shows that the HVS is unable to detect differences between objects and their background under certain conditions of luminance, distance, or spatial frequency [2–6]. Compression artifacts can be masked under these conditions because they function as foreground, while the scene acts as the background.

Texture and luminance masking are two techniques that also exploit properties of the HVS to reduce compression artifacts. Texture masking takes advantage of the fact that the presence of texture in some areas of the image can mask some of the reconstruction errors, making it more difficult to detect a compression artifact in a textured area than in a homogeneous one. On the other hand, luminance masking is based on the observation that compression artifact errors are less noticeable in areas with high or low luminance. This means that errors in dark or bright regions of an image are less visible to the HVS, allowing for the reduction in the amount of information to be encoded without significant perceptual loss.

The rest of this paper is structured as follows. In Section 2, the state of the art is presented. The proposed contrast and texture masking models for the HEVC video coding standard are explained in Section 3. Section 4 gives the results when masking techniques are applied to a series of well-known video sequences. Finally, Section 5 summarizes the conclusions of this study and makes some recommendations for future research.

2. Related Work

Tong et al. [7] proposed a perceptual model of texture and luminance masking for images compressed using the JPEG standard [8]. The authors provided a method to classify the transform blocks according to their content type, namely, texture blocks (containing a lot of spatial activity), edge blocks (containing a clear edge as primary feature) or plain blocks (generally smooth, with low spatial activity). The authors claimed that human sensitivity to error was, in general, inversely proportional to the spatial activity, and was extremely sensitive to low spatial activity areas (plain blocks). To perform this classification, the authors used an algorithm that was based on the weight of the Discrete Cosine Transform (DCT) coefficients grouped by their frequency or position within the transformed block. Finally, the degree of additional quantization that should be applied to each block was determined in such a way that the distortions produced by increments in quantization remained masked.

Tong's model has been modified and refined by other authors. For example, Zhang et al. [9] built a luminance model and block classifier using the mean of the DCT coefficients. Zhang et al. also considered the intra-band masking effect, which refers to the imperceptible error tolerance within the sub-band itself. In other words, a different quantization value is applied for each coefficient within the 8×8 block, depending on the block classification and the coefficient position in the block.

Most models are based on partitioning the image into 8×8 blocks [9–11], however Ma et al. [12] extended the classification algorithm to block sizes of 16×16 to adapt for higher image resolutions. Furthermore, the proposed classification model was performed in the pixel domain. This was based on the Canny edge detector and applied an adaptive quantization scheme that depended on the block size. The problem of edge detection algorithms lies in finding the optimal threshold value: choosing a low value causes very small edges to be detected, while choosing a high value skips important edges [13]. Several authors [14,15] used a 4×4 reduction of the classifier proposed in [7].

Regarding video coding standards, several studies have incorporated perceptual coding schemes in their reference software. In MPEG-2 Test Model 5 [16], a quantization parameter (QP) offset based on the spatial activity is defined, which is calculated as a function of the variance of pixels within the macroblock. Tang et al. [17] proposed a bit allocation technique for the JM7.6 reference software of the H.264/AVC video coding standard by adopting a novel visual distortion sensitivity model that was based on motion attention and texture structure.

From version 16 of the HEVC reference software encoder description [18], there has been an option called adaptive QP that varies the quantization parameter for each coding unit (CU) to provide improved perceptual image quality. This algorithm is based on the algorithm used in MPEG-2 TM5. Prangnell et al. [19] proposed a modified version of the adaptive QP algorithm by extending the spatial activity calculation to the chrominance channels and obtained better performance than when using only the luminance. In [20], Kim et al. designed a perceptual video coding scheme for HEVC based on Just Noticeable Differences (JND) models, including contrast, texture, and luminance masking, in both transform and pixel domains. JND models are based on determining the threshold under which the HVS is unable to perceive differences from the original source. The main drawback of [20] is that the behavior of the rate-distortion optimization (RDO-based) mode decision is modified, and therefore, corrective factors are required to compensate for the distortion introduced by JND.

Wang et al. [21] proposed a block-level adaptive quantization (BLAQ) for HEVC, where each CU had its own QP adapted to the local content. The authors did not use masking techniques to determine the QP; instead, it was obtained by a brute-force algorithm. To reduce the complexity of the algorithm, the authors modified the rate-distortion cost function that gives priority to the distortion, as measured in the Peak Signal-to-Noise Ratio (PSNR). Xiang et al. proposed in [22] a novel adaptive perceptual quantization method based on an adaptive perceptual CU early-splitting algorithm to address the spatial activity and Lagrange multiplier selection problems in the HEVC quantization method. In [23], Zhang et al. proposed a method to predict the optimal QP value at the Coding Tree Unit (CTU) level by employing spatial and temporal combined masks using the perception-based video metric (PVM). Because the default CTU block size is 64×64, this work did not take advantage of HEVC's quadtree partitioning when applying masking techniques in smaller regions.

Recent advancements in the development of contrast masking models using deep learning have been reported in literature. Marzuki et al. [24] proposed an HEVC perceptual adaptive quantization based on a deep neural network. They determined the QP at the frame level and therefore did not take advantage of texture masking in scenes with multiple texture types. Bosse et al. [25] proposed a method of distortion-sensitive bit allocation in image and video compression based on distortion sensitivity estimated using a deep Convolutional Neural Network (CNN). Sanagavarapu et al. [26] explored the use of Region of Interest (ROI) techniques by segmenting the surgical incision region and encoding it with the complexity-efficient scalable HEVC, highlighting the application of perceptual algorithms to improve bit rate efficiency while maintaining visual quality in surgical telementoring systems.

An important aspect to be considered when including masking in an encoder is the way that the block type or the adaptive quantization value to be applied in each block is signaled in the bitstream. Most of the cited authors use the thresholds that are defined by the JND model to discard the coefficients below a certain value (i.e., being included in the image or video encoding algorithm) without sending additional information to the decoder. In [7], Tong et al. analyzed the performance of both methods, namely the first method that does not send extra information and the second method that requires extra side information to be sent to the decoder. They concluded that the latter method achieved a better rate-distortion (RD) performance. Studies that are based on modifying the QP

value at the slice or block level often make use of the delta QP parameter, which is the difference between the current QP and the previously encoded QP.

Many of the works that have been cited so far make use of the PSNR distortion metric to evaluate the RD performance. However, it is well-known that the PSNR metric does not accurately reflect the perceptual assessment of quality [27,28]. Consequently, in studies, such as [12], subjective tests were conducted using the Difference Mean Opinion Score (DMOS) as an indicator of perceptual quality. However, given that the PSNR is not an adequate metric to properly evaluate the impact of perceptual techniques, we decided to use some objective metrics that attempt to characterize the subjectivity of the HVS, such as Structural Similarity (SSIM) [29], Multi-Scale SSIM (MS-SSIM) [30], and PSNR-HVS-M [31], to measure their RD performance.

The SSIM and the MS-SSIM metrics are based on the hypothesis that the HVS is highly adapted to extract structural information from the scenes. Both metrics consider the luminance, contrast, and structural information of the scenes, whereas MS-SSIM also considers the scale. The PSNR-HVS-M metric, which is a modified version of PSNR, considers the contrast sensitivity function and the between-coefficient contrast masking of DCT basis functions.

In this work, we present a novel scheme of texture and contrast masking to be applied in the HEVC reference software [18]. For the contrast masking model, we start from the frequency-dependent quantization matrices that are included in the HEVC reference software for blocks from 8×8 to 32×32 sizes. In addition, we add a new 4×4 weighting matrix [32] that achieves an additional rate reduction while maintaining the perceptual quality. For the texture masking model, we make use of the mean directional variance (MDV) metric, and we use a support vector machine (SVM) to perform the block classification (plain, edge, or texture). The QP offset value is calculated as a function of the block classification and its texture energy, in a similar way as that proposed by Tong et al. [7].

To demonstrate the potential of this novel scheme, we encode a set of well-known test sequences and analyze their performance in terms of rate and distortion. The results are presented with the BD-rate model [33], using the SSIM, MS-SSIM, and PSNR-HVS-M distortion metrics.

The main innovations provided by this work are the following ones:

- An improved contrast masking method that covers all HEVC available block sizes (4×4 to 32×32) that includes a new efficient quantization matrix;
- A new block classification method for block texture masking based on the MDV metric that efficiently classifies every block as a texture, edge, or plain block;
- A new QP offset calculator for the HEVC adaptive QP tool, based on the block texture energy and its classification.

All these innovations define a novel perceptual quantizer based on the one proposed in the HEVC reference software.

3. Proposed HEVC Perceptual Quantizer

In this section, the details of the new perceptual quantizer for the HEVC video coding standard is described. We first describe how CSF masking is applied in HEVC (scaling list tool) followed by the proposed improvements. Then, after applying the CSF masking, we use a texture masking over-quantization scheme that is based on (a) the use of a new block classifier, and (b) an optimized over-quantizer that depends on the block type and its energy.

3.1. Proposed Contrast Sensitivity Function

Contrast masking is a perceptual compression technique that exploits the visual adaptation of the HVS to contrast. This adaptation depends on the amount of contrast between an object and its surroundings (or background), the distance, and the spatial frequency. Several studies have been performed to characterize the CSF using subjectively measured human contrast thresholds for different spatial frequencies [3,5,6]. In this regard,

the Mannos and Sakrison model [2] and the Daly model [4] are among the most popular in the field of image and video coding.

The HEVC standard uses a frequency-dependent quantization to implement CSF masking [34]. Depending on its contrast sensitivity importance, a different amount of quantization is applied to each frequency coefficient of a block (i.e., the higher the perceptual importance, the lower the corresponding quantization level).

With this goal in mind, the HEVC reference software defines the use of static non-uniform quantization matrices, which are also called weighting matrices, by setting the `ScalingList` (SCL) option to 1 (default is 0) in the coding configuration parameters. These weighted quantization matrices are defined for the intra- and interpredictions, as well as for the luminance component and the chrominance components. In terms of CUs, non-uniform quantization matrices are only defined for CUs of 8×8 (see Figure 1b). Meanwhile, for 16×16 and 32×32, the matrices are obtained by upsampling, using a replication of the 8×8 matrix. In the case of 4×4 CUs, the HEVC reference software does not define any weighting matrix, and therefore it uses a uniform matrix (see Figure 1a).

$$\begin{bmatrix} 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 \end{bmatrix} \qquad \begin{bmatrix} 16 & 16 & 16 & 16 & 17 & 18 & 21 & 24 \\ 16 & 16 & 16 & 16 & 17 & 19 & 22 & 25 \\ 16 & 16 & 17 & 18 & 20 & 22 & 25 & 29 \\ 16 & 16 & 18 & 21 & 24 & 27 & 31 & 36 \\ 17 & 17 & 20 & 24 & 30 & 35 & 41 & 47 \\ 18 & 19 & 22 & 27 & 35 & 44 & 54 & 65 \\ 21 & 22 & 25 & 31 & 41 & 54 & 70 & 88 \\ 24 & 25 & 29 & 36 & 47 & 65 & 88 & 115 \end{bmatrix}$$

(**a**) Default 4×4 quantization weights (intra- and interprediction)

(**b**) Default 8×8 quantization weights (intraprediction)

Figure 1. Default HEVC quantization weighting matrices.

In this work, we include a new 4×4 weighting matrix to increase the compression level for small blocks while maintaining the same perceptual quality. Instead of deriving the matrix weights by downsampling the default quantization matrix of size 8×8, as the standard does for the higher-resolution matrices, we propose to determine the weights of the 4×4 matrix from the study presented in [32]. The author proposes the use of the CSF model of Daly [4] (Equation (1)), where f is the radial frequency in cycles/degree (cpd), assuming the best viewing conditions in which defects are detected earlier. In other words, using a high-resolution display and a short viewing distance. Coding or compression defects may be masked by the content and by the visual capacity at higher resolution and longer viewing distance.

$$H(f) = 2.2(0.192 + 0.114 \cdot f) \cdot e^{-(0.114 \cdot f)^{1.1}} \tag{1}$$

In order to determine the maximum frequency represented in the signal (f_{max}), we begin by calculating the sampling frequency (f_s) using Equation (2). The maximum frequency is then given by Equation (3).

$$f_s = \frac{v \cdot \tan(1^\circ) \cdot r}{0.0254} \tag{2}$$

$$f_{max} = \frac{f_s}{2} \tag{3}$$

Assuming a display resolution of $r = 600$ pixels per inch (ppi) and a viewing distance $v = 12.23$ inches, the maximum frequency is $f_{max} = 64.04$. The CSF curve obtained with Equation (1) is shown in Figure 2.

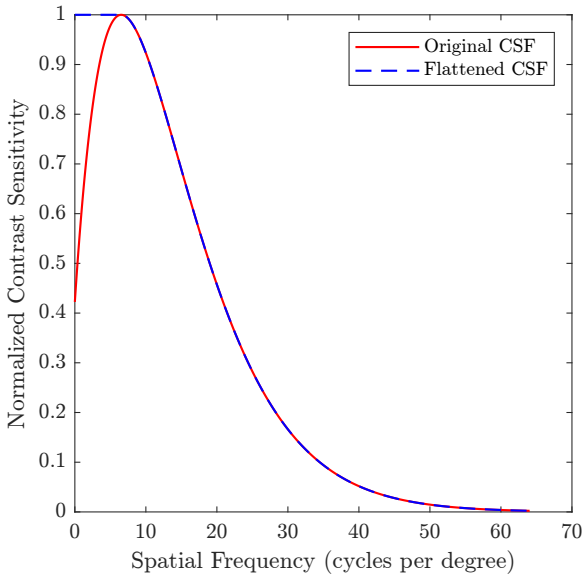


Figure 2. Contrast sensitivity function. The red curve represents the original CSF as defined by Equation (1), while the blue dashed curve represents the flattened CSF, with spatial frequencies below the peak sensitivity saturated.

The red curve corresponds to the definition of the CSF according to Equation (1). As we can see, the HVS is most sensitive in an intermediate region, acting as a bandpass filter, while it is less sensitive to very low and very high frequencies. In addition, as shown with the blue dashed curve in Figure 2, spatial frequency values below the maximum sensitivity peak have been saturated. This is done to preserve the information of the coefficients close to the DC component and including it, because it is in that region where most of the information (energy) is concentrated after applying the DCT to a block.

Using the CSF model of Daly [4], the CSF curve is calculated as shown in Equation (1). This curve represents the sensitivity of the HVS to different spatial frequencies. Each coefficient in the 4×4 DCT block corresponds to a specific spatial frequency. The frequency $f(u, v)$ for each coefficient (u, v) is calculated by considering the horizontal and vertical frequencies of the DCT basis functions. The radial frequency $f(u, v)$ is given by Equation (4).

$$f(u, v) = \sqrt{u^2 + v^2} \tag{4}$$

where $u, v \in 0, 1, 2, 3$. The calculated frequencies are then mapped onto the CSF curve to get the sensitivity values. These values represent how sensitive the HVS is to the corresponding frequencies in the DCT block. The sensitivity values are scaled and normalized to obtain the final weighting values. The scaling ensures that the weights are appropriately adjusted to maintain perceptual quality while increasing compression efficiency. The normalization step involves scaling the values such that the smallest value is 16 and the largest value is scaled to match the highest weight used in the HEVC standard matrices. Finally, the proposed 4×4 weighting matrices are obtained (Figure 3).

$$\begin{bmatrix} 16 & 16 & 20 & 32 \\ 16 & 17 & 21 & 37 \\ 20 & 21 & 29 & 55 \\ 32 & 37 & 55 & 115 \end{bmatrix} \qquad \begin{bmatrix} 16 & 16 & 19 & 29 \\ 16 & 17 & 20 & 32 \\ 19 & 20 & 26 & 46 \\ 29 & 32 & 46 & 91 \end{bmatrix}$$

(**a**) Intraprediction (**b**) Interprediction

Figure 3. Proposed 4×4 quantization weighting matrices for intra- and interprediction modes.

For the remaining block sizes, we use the default weighting matrices that were proposed by the HEVC standard. To implement our proposal in the HEVC reference software, we set the ScalingList parameter to 2. This allows us to define a custom weighting matrix scheme from a text file, which is identified by the ScalingListFile parameter.

To measure the impact of this optimization, we conducted an experimental test using the HEVC reference software version 16.20 [35]. The test video sequences (see Table 1 and Appendix A) from the HEVC conformance test proposal [36] were encoded with the SCL parameter set to 1 (default weighting matrices) and 2 (custom weighting matrix scheme), and the gains (BD rate) were obtained and compared to the default encoding (SCL set to 0). The other coding tools were left with their default values, with the exception of the transform skip (TransformSkip) parameter, which was disabled for all sequences except those of class F to maximize the perceptual response, as stated in [37].

Table 1. HEVC video test sequence properties.

Class	Sequence Name	Resolution	Frame Count	Frame Rate	Bit Depth
A	Traffic	2560 × 1600	150	30	8
	PeopleOnStreet		150	30	8
	Nebuta		300	60	10
	SteamLocomotive		300	60	10
B	Kimono	1920 × 1080	240	24	8
	ParkScene		240	24	8
	Cactus		500	50	8
	BQTerrace		600	60	8
	BasketballDrive		500	50	8
C	RaceHorses	832 × 480	300	30	8
	BQMall		600	60	8
	PartyScene		500	50	8
	BasketballDrill		500	50	8
D	RaceHorses	416 × 240	300	30	8
	BQSquare		600	60	8
	BlowingBubbles		500	50	8
	BasketballPass		500	50	8
E	FourPeople	1280 × 720	600	60	8
	Johnny		600	60	8
	KristenAndSara		600	60	8
F	BaskeballDrillText	832 × 480	500	50	8
	ChinaSpeed	1024 × 768	500	30	8
	SlideEditing	1280 × 720	300	30	8
	SlideShow		500	20	8

The average BD-rate performance (negative values mean gains) for different perceptual metrics is shown in Table 2. Low BD-rate gains were achieved (always below 1%) by enabling only the weighting matrices included in the HEVC standard (SCL = 1). Even for low-resolution sequences (classes C and D), BD-rate losses were observed for some metrics, such as for SSIM metric in class D sequences, where a loss of 1.26% was introduced.

Table 2. Average coding performance [% BD rate] when using our proposed 4 × 4 weighting matrix (intraprediction).

Sequence Class	SCL = 1 (HEVC Presets)			SCL = 2 (Ours)		
	SSIM	MS-SSIM	PSNR- HVS-M	SSIM	MS-SSIM	PSNR-HVS-M
Class A	−0.66	−0.33	−0.62	−1.06	−0.82	−1.58
Class B	−0.97	−0.48	−0.99	−3.20	−2.58	−4.23
Class C	0.26	0.08	−0.08	−4.82	−5.36	−7.39
Class D	1.26	0.29	−0.05	−1.36	−5.66	−7.65
Class E	−0.74	−0.50	−0.75	−1.78	−1.39	−1.98
Class F	−0.15	−0.04	−0.11	−4.57	−4.19	−4.17
Average	−0.17	−0.16	−0.43	−2.80	−3.33	−4.48

As shown in Table 2 (SCL = 2), our proposal obtained a remarkable increase in BD-rate gains for all cases. The improvement was between 2.64% and 4.05% on average for all classes. The SSIM metric scores were lower when compared to the other metrics on low-resolution video sequences (classes C and D), while PSNR-HVS-M obtained the highest BD-rate gains (above 7.39%) for these sequences. Meanwhile, there seemed to be a consensus on all metrics for class E (video-conference applications) and F (synthetic or artificial) sequences because they all obtained broadly similar results.

In Figure 4, we can see that our proposal reduced the bit rate considerably as the quantization parameter decreased, in other words, at low compression rates. This occurred because as the value of the quantization parameter was reduced, the number of TUs (transform units) of size 4 × 4 increased, and thus, the performance impact of our proposed 4 × 4 weighting matrix was more noticeable.

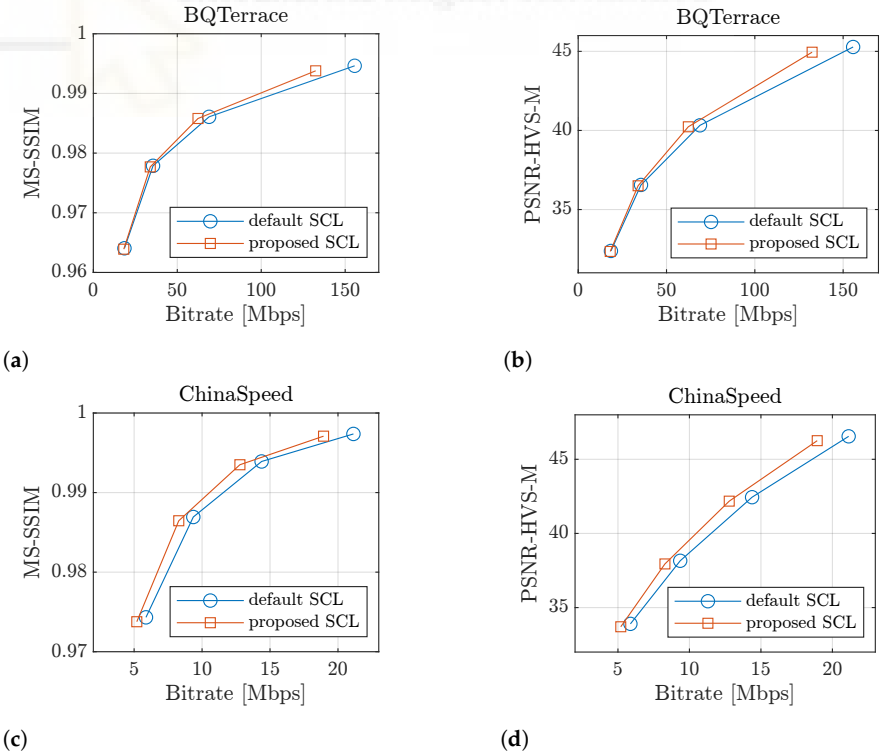


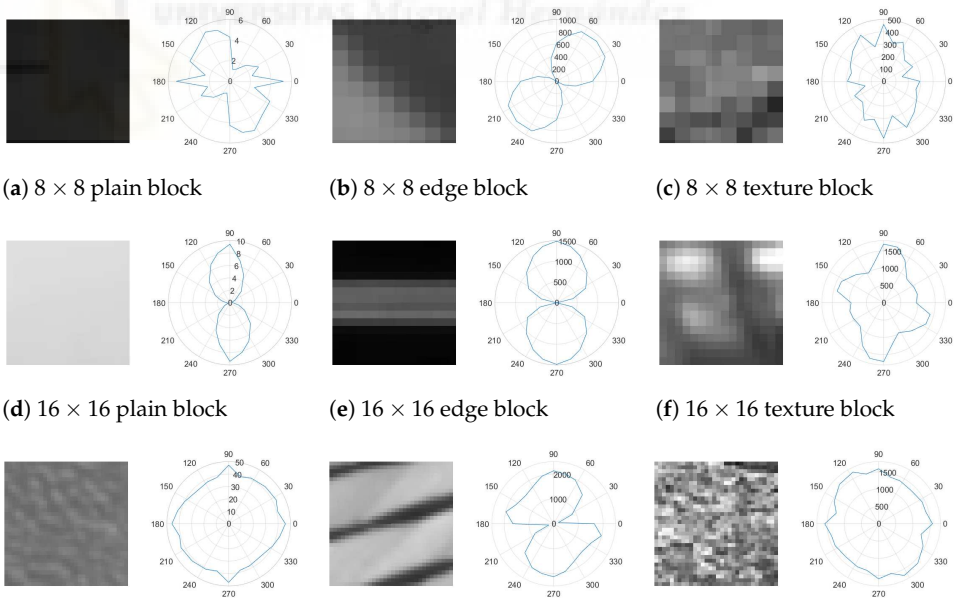
Figure 4. Rate-distortion curves comparing our proposed CSF with the default implemented in the HEVC standard using different perceptual metrics. (**a**,**b**) correspond to the BQTerrace sequence of class B, while (**c**,**d**) correspond to the ChinaSpeed sequence of class F.

3.2. Block Classification Based on Texture Orientation and SVM

After applying the improved CSF masking, we proceed to compute the proper QP offset based on the block texture information. For this purpose, we first need to identify the texture info of each block by means of a block classifier in a similar way to what Tong et al. [7] proposed for the JPEG image encoder. In [7], the authors stated that to maximize perceptual RD, plain blocks should not be over-quantized; the edge blocks could be minimally over-quantized and texture blocks could be over-quantized according to their texture energy level.

The main limitation when importing the texture classifier scheme that was proposed by Tong et al. into the HEVC standard is the adaptation to the different block sizes. JPEG only uses 8 × 8 block size, whereas HEVC includes a wide variety of CU sizes. It should also be considered that the HEVC standard uses the integer transform (I-DCT and I-DST) of the prediction residual. For those reasons, we propose a novel texture block classification using a supervised SVM, which uses the features obtained from the MDV metric proposed by Damian et al. [38] as input features.

Our first step was the classification of about 1800 HEVC-encoded luma blocks of different sizes, depending on whether they were smooth, edged, or textured. To achieve this, we randomly selected blocks from some image databases, such as the ESPL Synthetic Image Database [39], USC-SIPI Image Database [40], TESTIMAGE [41] and Kodak image dataset [42]. To avoid bias in human classification, five different video coding researchers participated in the classification process. The users classified the blocks according to their type (texture, plane, or edge) by using software that randomly presented the blocks for classification. As an example, Figure 5 shows several manually classified blocks that are organized according to size and block type. As can be seen, the blocks that were classified as plain have a smooth content. In contrast, the content of the texture blocks exhibits a more random pattern. The blocks classified as edge have a very pronounced directionality.



(**a**) 8 × 8 plain block (**b**) 8 × 8 edge block (**c**) 8 × 8 texture block

(**d**) 16 × 16 plain block (**e**) 16 × 16 edge block (**f**) 16 × 16 texture block

(**g**) 32 × 32 plain block (**h**) 32 × 32 edge block (**i**) 32 × 32 texture block

Figure 5. Samples of manually classified blocks (left-hand side) and their associated polar diagram of the MDV metric (right-hand side). From top to bottom: 8 × 8, 16 × 16, and 32 × 32 block sizes; from left- to right-hand side: plain, edge, and texture blocks.

Figure 5 also shows the polar diagram of the MDV values for each block. This metric measures the local directionality of an image by calculating the cumulative variance along discrete lines in the given direction. Using the version of MDV that was introduced in [38],

we computed the twelve rational slopes of all the manually classified blocks to find any correlation between the values of this metric and the classification result. Because the 4×4 block size did not provide sufficient resolution to calculate the 12 rational slopes, and even the manual classification performed by human observers was not completely coherent, the 4×4 blocks were discarded from the texture over-quantization process.

Interesting results can be extracted from the experiments and results shown in Figure 5. On the one hand, texture blocks tended to exhibit polar diagrams that were close to circular shapes, which showed high variance values in all directions. However, edge blocks had a minimum (dominant gradient) in the direction of the edge orientation. Strong edges in a block had higher differences between the minimum and maximum MDV values and were used to form a polar diagram with an "8" shape. Plain blocks tended to have a variety of patterns; however, all of them had relatively very low MDV values when compared to texture and edge blocks (see Figure 5a,d,g).

To establish a robust block classification, we decided to use an SVM classifier. A SVM is a machine learning technique that facilitates linear and non-linear binary classification. Because we wanted to get three block clusters (plane, edge, and texture), we had to use either of two multi-class methods: One vs. One (OvO) or One vs. Rest (OvR). The main difference between these two techniques lies in the number of binary classifier models required. In the OvR strategy, the multi-class classification is split into one binary classification model per class, while for the OvO strategy, for the N-class instances dataset, $(N(N-1))/2$ binary classification models are needed. Because we had only three clusters, both techniques required the same number of binary classification models, and therefore both strategies had similar computational costs.

After analyzing the results of applying different statistics to the MDV data (e.g., the mean, the variance, the median, etc.), it was observed that the best results (i.e., a better clustering in the $\mathbb{R}^3$ space) were obtained using the mean, the variance, and the minimum value of the MDV as the input features to be used in the SVM algorithm. The manual classification of 16×16 blocks of the training dataset is shown in Figure 6a. Texture occupies the YZ plane (they have low $var(MDV)$), edge blocks occupy the XY plane (they have low $min(MDV)$), and plain blocks stay close to the origin of coordinates.

Given that the available block sizes in the HEVC standard are limited, instead of using the block size as an additional feature of a single SVM model, we decided to use one SVM model for each block size.

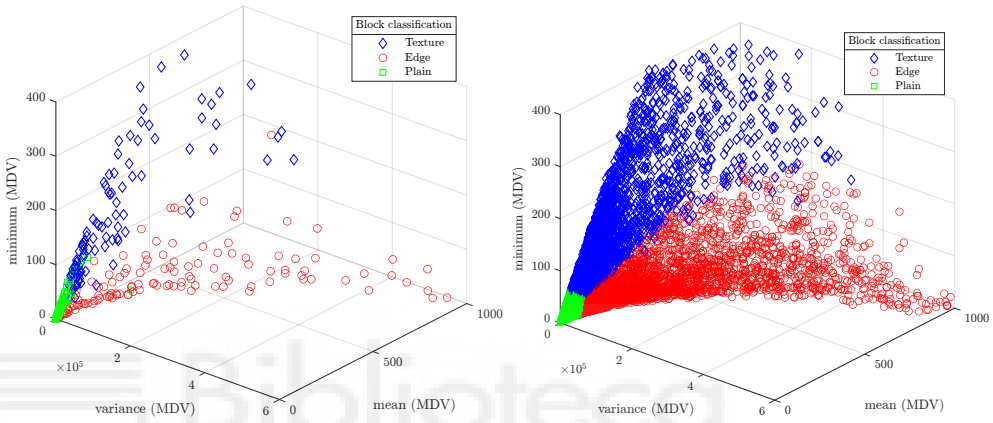
SVM models were implemented and trained using the Classification Learner application from MATLAB R2020a. The optimizable support vector machine was selected to find the optimal SVM model parameters, including kernel function type (linear, quadratic, cubic, or Gaussian), kernel scale, box constraint, and multi-class method (OvO and OvR).

The optimal parameters and resulting model accuracy of the three models (after 30 iterations of Bayesian optimization) are shown in Table 3. As can be seen, a high degree of accuracy was obtained for all the models, which was sufficient for correct block classification. Figure 6b shows the classification of 16×16 blocks belonging to the testing dataset. It can be seen that the model properly classified the blocks into texture, edge, or plain.

As a visual example, Figure 7 shows the result of applying block classification to the CUs of a BasketballDrill frame quantized at QP 32. It can be seen that the lines of the basket court were correctly labeled as edge blocks, while some parts of the basket net were considered as texture blocks.

Table 3. Optimized SVM models: parameters and accuracy.

Model Parameters	Block Size		
	8 × 8	**16 × 16**	**32 × 32**
Kernel function	linear	linear	linear
Kernel scale	auto	auto	auto
Box constraint level	85	285	35
Multi-class method	One-vs.-All	One-vs.-One	One-vs.-All
Standardize data	true	true	true
Model accuracy	93.9%	95.4%	94.5%



(**a**) Training dataset (**b**) Testing dataset

Figure 6. (**a**) Scatter plot of manually classified 16 × 16 blocks (training dataset), and (**b**) the classification results provided by the trained SVM model (testing dataset)

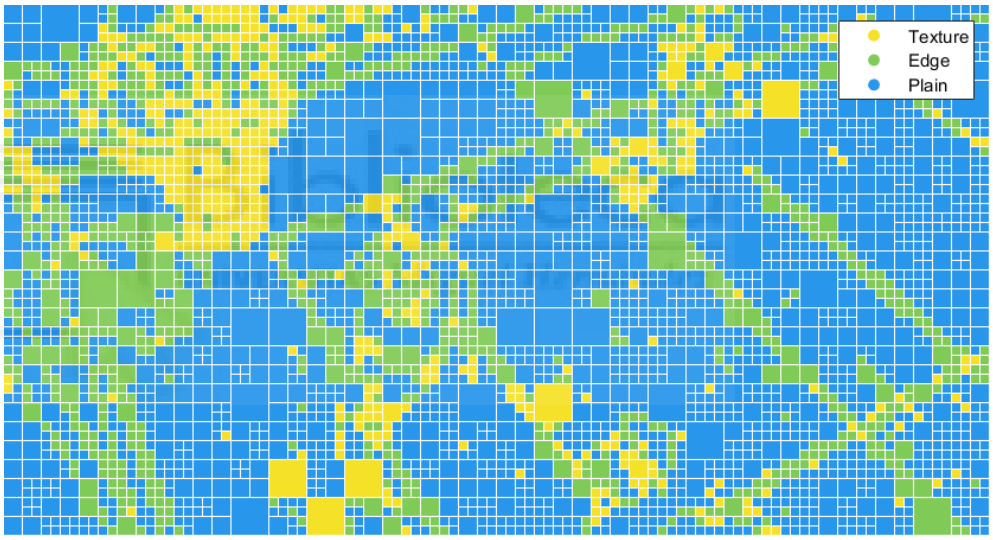
To integrate the trained SVM models into the HEVC reference software (HM) for evaluation, we exported the trained SVM models from MATLAB to C++. In the HM code, block classification is computed at the frame level before the quadtree partitioning and RDO stage, similar to the adaptive QP algorithm of the HEVC video coding standard [18]. The SVM model inference was performed using the exported C++ code to ensure compatibility and efficiency within the HM framework.

The workflow of the block classification code is as follows: after loading and storing the original YUV pictures into the picture buffer list, if texture masking is enabled, then the function `xPreanalyzeTextureMasking` is called. This function splits each frame into square blocks of size 32, 16, and 8 pixels, the classification of each one is calculated using the corresponding SVM model according to its size. The result is stored in memory. It also calculates and stores the block energy (ε) (defined in Section 3.3), which is required to compute the over-quantization (QP offset). Later, during the partitioning and RDO stage, the block type and energy of each CU are already available according to its size and location inside the frame.

3.3. Obtaining optimal QP offset

The next step after classifying a CU block is to obtain its optimal QP offset. We defined the block energy (ε) as the absolute sum of all of the AC-transformed coefficients of the original picture. The energy distribution was analyzed according to the block type (texture, edge, or plain) and its size. In Figure 8, the block energy distribution is shown as a box plot for each block size and type. This representation allowed us to graphically visualize the five-number summary, which consisted of the minimum and maximum range values, the upper and lower quartiles, and the median.

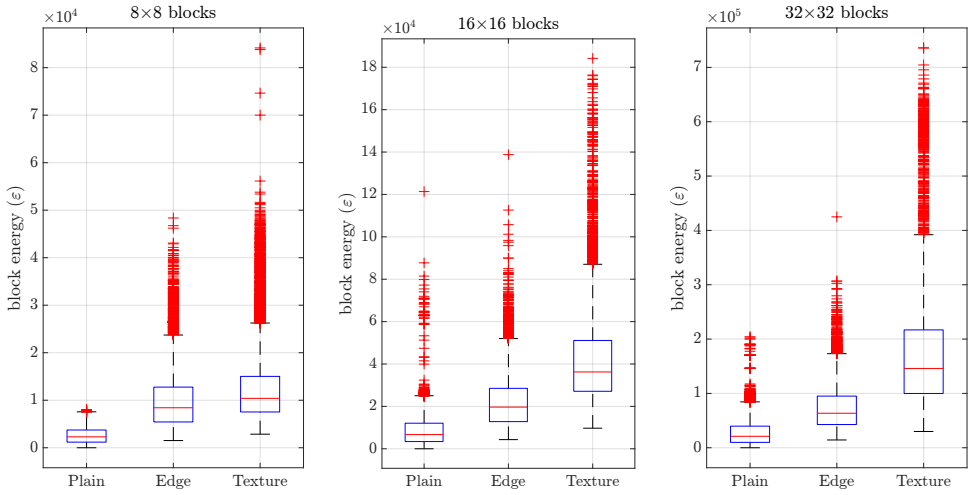
(**a**) Original BasketballDrill frame



(**b**) Block classification using QP = 32

Figure 7. Example of block classification for the first frame of sequence BasketballDrill, using optimal SVM models for each block size.

A pattern can be observed in terms of the block energy distribution according to the block classification. As expected, blocks classified as texture have the highest block energy distribution, followed by edge blocks and finally, plain blocks have the lowest energy distribution. The outliers in Figure 8 result from synthetic, computer-generated sequences, which exhibit high energy in the middle and high bands. These differ from the majority of blocks from natural sequences in our dataset, explaining the appearance of these extreme cases as outliers.



(**a**) 8 × 8 block size (**b**) 16 × 16 block size (**c**) 32 × 32 block size

Figure 8. Box and whisker plot of the block energy (ε) distribution by size and texture classification.

$$\Delta QP_{i,j} = \left\lfloor \frac{6 \cdot \ln(QStep_{i,j})}{\ln(2)} \right\rfloor \tag{5}$$

$$QStep_{i,j} = \begin{cases} 1 & \text{if } \varepsilon(B_{i,j}) \leq MinE, \\ MaxQStep & \text{if } \varepsilon(B_{i,j}) \geq MaxE, \\ 1 + \frac{MaxQStep-1}{MaxE-MinE} \times & \\ (\varepsilon(B_{i,j}) - MinE) & \text{otherwise} \end{cases} \tag{6}$$

In the HEVC standard, the adaptive QP mode assigns to each CU a QP offset or ΔQP that modifies the slice QP adaptively by means of a rate-distortion analysis where PSNR is the distortion metric. Our objective was to also obtain a ΔQP for each CU but we followed a different approach based on the block energy. The distortion metric that we used was perceptually based (e.g., SSIM, MS-SSIM, or PSNR-HVS-M metric).

Equation (5) shows the inverse procedure to obtain ΔQP, as proposed in [43], where $QStep_{i,j}$ is the quantization step size for the CU block $B_{i,j}$ in the block partitioning map, and $\Delta QP_{i,j}$ is the QP offset parameter to be applied to over-quantize the $B_{i,j}$ block. When $QStep_{i,j} = 1$, then $\Delta QP_{i,j} = 0$ (i.e., no additional quantization should be applied to the $B_{i,j}$ block).

To obtain the $QStep_{i,j}$ value for a block, we used the linear threshold elevation function that was presented in Equation (6), similarly to the one proposed in [7], where $MaxE$ and $MinE$ correspond to the maximum and minimum block energy of the set of blocks belonging to the same block type and size (Figure 8), $MaxQStep$ is the maximum allowed quantization step size, $\varepsilon(B_{i,j})$ is the energy of the current block $B_{i,j}$, and $QStep_{i,j}$ corresponds to the quantization step to be assigned to the block. Figure 9 shows the representation of Equation (6), where the two lines show how the slope of the function varies for two different sets of function parameters (i.e., $MinE$, $MaxE$, and $MaxQStep$). As we can see in Figure 9, the corresponding $QStep_{i,j}$ is different for each parameter set, while the block energy $\varepsilon(B_{i,j})$ is the same. The question that arises here is how to choose the function parameters to maximize the overall BD rate [33] performance value. The BD rate was computed by considering the use of a perceptual distortion metric instead of the PSNR.

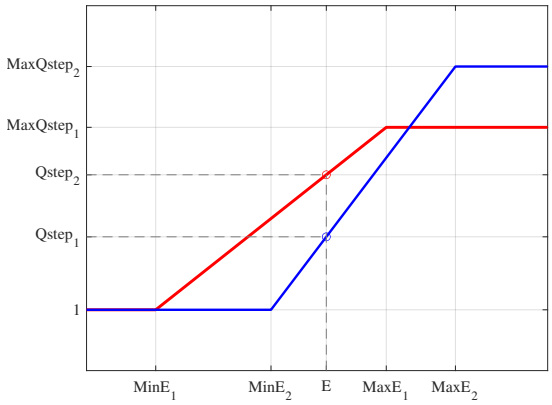


Figure 9. Representation of Equation (6) for two sets of function parameter, (**red**) $MinE_1$, $MaxE_1$ and $MaxQStep_1$ and (**blue**) $MinE_2$, $MaxE_2$, and $MaxQStep_2$. $\Delta QStep_{i,j}$ is different for each set.

We used different sets of parameters to find the optimum combination for each block size (i.e., 8×8, 16×16 and 32×32) and for each block type (i.e., texture and edge). We did not consider plain blocks because they are more sensitive to visible artifacts [7] and should not be over-quantized.

Figure 10 summarizes all the tested parameter sets for each block size and type. A parameter set is built by following the connection arrows in the graph. For example, in the first set, $MinE$ receives the value of the energy at the lower whisker (i.e., 0th percentile), $MaxE$ receives the energy at the bottom of the box (i.e., 25th percentile), and finally the value 1.1 is given to $MaxQStep$. The second parameter set has the same values for $MinE$ and $MaxE$, but we change $MaxQStep$ to 1.2, and so on. To guarantee that the range of the resulting $\Delta QP_{i,j}$ is bounded between 0 and 7 (maximum QP offset allowed in HEVC), we restricted the $MaxQStep$ range to be between 1.1 and 2.5.

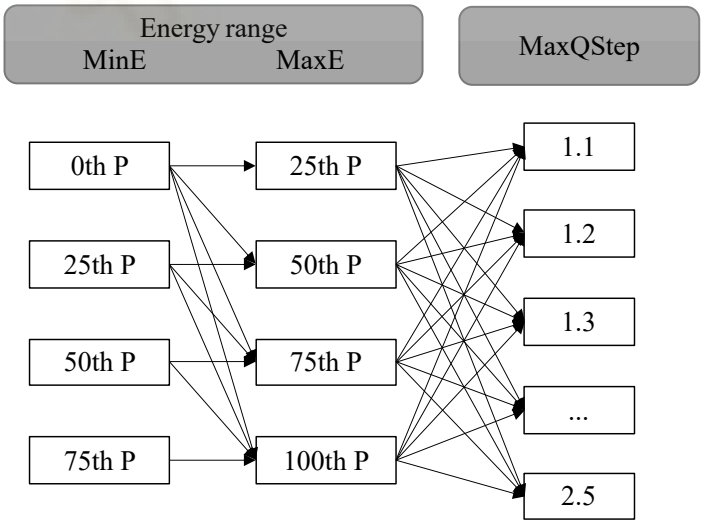


Figure 10. Flowchart of candidate selection for brute-force analysis of perceptually optimal parameters. The Ps in energy range boxes refer to the percentile.

We used the BD rate [33] as a performance metric to determine the best parameter set. Therefore, for each one, we ran a set of encodings using QP values 22, 27, 32, and 37 with the video test sequences belonging to classes A, B, and E which had the highest frame resolution (as suggested in [36]).

After collecting all of the results, we determined the near optimal *MaxE*, *MinE*, and *MaxElevation* values for each block type and size, as in Table 4.

Table 4. Optimal linear function parameters.

Classification	Parameter	Block Size		
		8 × 8	**16 × 16**	**32 × 32**
Texture	MinE	2864	9712	29,952
	MaxE	26,256	26,800	216,880
	MaxElevation	1.3	1.2	2.2
Edge	MinE	1520	4320	14,320
	MaxE	5424	52,016	63,504
	MaxElevation	1.2	1.3	1.2

As an example, applying the optimum parameter set for texture blocks of size 8 × 8 in the PeopleOnStreet video test sequence is shown in Figure 11. This figure shows the evolution of the BD rate (lower is better) for different values of the *MaxQStep* parameter. Each curve corresponds to a certain block energy range (*MinE* and *MaxE* parameters). It can be seen that, for this particular case, the energy range from the 0th to 25th percentile (purple curve with circle marks) obtains the highest BD-rate gain when $MaxElevation = 1.3$.

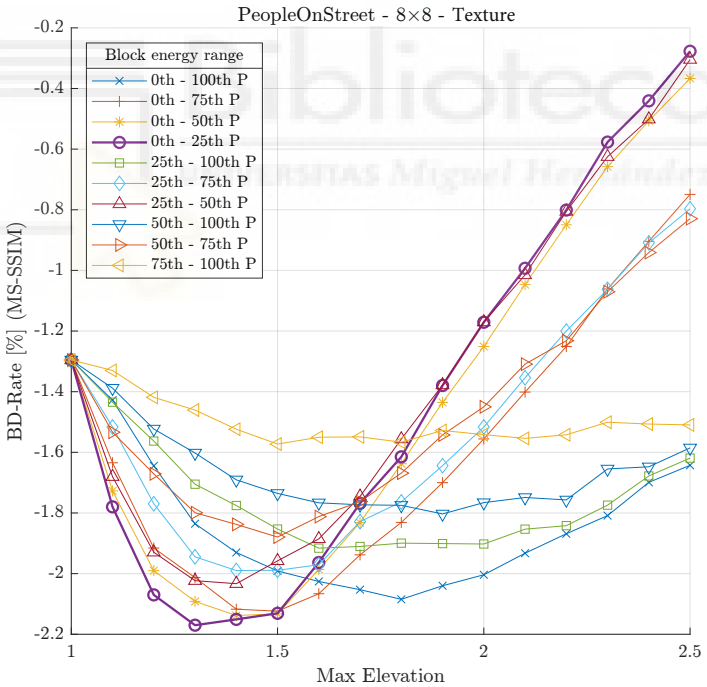


Figure 11. BD-rate curves (MS-SSIM metric) for PeopleOnStreet video test sequence over the *MaxQStep* parameter when modifying texture blocks of size 8. Each curve represents a different block energy range (*MinE* and *MaxE*).

The BD-rate performance for all of the objective quality metrics used after applying the optimal parameters is shown in Table 5. Each column shows the results of applying the optimal over-quantization values only to blocks of the corresponding block type and size.

Table 5. Average coding performance [% BD rate] after applying the optimal $\Delta QP_{i,j}$ values derived from our texture masking proposal.

Class	Metric	Texture Blocks			Edge Blocks		
		8 × 8	16 × 16	32 × 32	8 × 8	16 × 16	32 × 32
A	SSIM	−1.04	−0.98	−1.01	−0.67	−1.07	−1.05
	MS-SSIM	−0.87	−0.76	−0.80	−0.46	−0.80	−0.82
	PSNR-HVS-M	−1.69	−1.44	−1.52	−1.26	−1.52	−1.57
B	SSIM	−3.74	−3.14	−3.15	−3.03	−3.21	−3.19
	MS-SSIM	−3.02	−2.47	−2.52	−2.34	−2.56	−2.57
	PSNR-HVS-M	−4.58	−4.05	−4.17	−3.90	−4.16	−4.21
E	SSIM	−2.12	−1.74	−1.77	−1.48	−1.87	−1.78
	MS-SSIM	−1.68	−1.35	−1.40	−0.98	−1.50	−1.39
	PSNR-HVS-M	−2.14	−1.89	−1.96	−1.17	−2.02	−1.99

4. Results and Discussion

To analyze the behavior of our HEVC perceptual quantizer proposal as a whole, we performed an exhaustive evaluation of the contrast and texture masking models that were described in the previous sections. Following the recommendations defined in the HEVC conformance test [36], we employed (a) all video test sequences proposed, grouping the results by the classes they belonged to (see Table 1) and (b) the BD-rate metric [33] using the SSIM, MS-SSIM, and PSNR-HVS-M as the perceptual video quality metrics. QP values of 22, 27, 32, and 37 were used to compute the BD rate.

The implementation of our proposed contrast and texture masking models was deployed using the HEVC reference software version 16.20 [35], running on a high-performance Linux server with an x86_64 architecture. The server was powered by two Intel® Xeon® Gold 6140 CPU @ 2.30GHz, each with 18 cores. The system was equipped with 376 GB of RAM.

To make texture masking compliant with the HEVC standard (in other words, to make the resulting bitstream readable with any HEVC-compliant decoder), we signaled the corresponding QP offset values at the CU level because the HEVC standard allows the transmission of a delta QP value for each CU block, that is, the difference in QP steps relative to the slice of QP that it belongs to [43].

Tables 6–8 show the results after encoding the whole set of video test sequences for all intra-, random-access, and low-delay coding configurations, respectively. In these tables, the "Contrast masking" column shows the gains that were obtained by applying only our CSF proposal presented in Section 3.1, while the "Contrast and Texture masking" column shows the total gains wgen applying the CSF and texture masking proposals, as explained in Section 3.3.

As expected, applying both contrast and texture masking techniques gave higher gains than applying contrast masking alone. For both of the structural information-based metrics (i.e., SSIM and MS-SSIM), the difference between using or not using texture masking implied an average BD-rate reduction of 1.92% for all intra- (AI), 3.02% for random-access (RA), and 3.44% for low-delay (LD) configurations. Regarding the PSNR-HVS-M metric, the benefit achieved by adding texture masking scheme was lower, with an average BD-rate reduction of 0.82%, 1.64%, and 1.91% for AI, RA, and LD, respectively. It seems that this metric does not take into special consideration the effect of texture masking generated by over-quantizing blocks with higher energy.

Table 6. Average coding performance in all of the intra-configurations [% BD rate].

Class	Sequence Name	Contrast Masking			Contrast and Texture Masking		
		SSIM	MS-SSIM	PSNR-HVS-M	SSIM	MS-SSIM	PSNR-HVS-M
A	Traffic	−1.00	−0.93	−1.77	−2.25	−1.89	−2.05
	PeopleOnStreet	−1.23	−1.27	−1.95	−3.38	−2.98	−2.54
	Nebuta	−1.22	−0.39	−1.64	−2.40	−1.70	−1.85
	SteamLocomotiveTrain	−0.80	−0.67	−0.98	−0.05	−0.04	−0.36
	Average	−1.06	−0.82	−1.58	−2.02	−1.65	−1.70
B	Kimono	−0.50	−0.41	−0.89	−0.53	−0.35	−0.81
	ParkScene	−2.26	−1.67	−3.11	−3.82	−2.91	−3.75
	Cactus	−2.97	−2.26	−4.06	−5.10	−3.94	−4.83
	BQTerrace	−6.68	−5.44	−7.82	−9.61	−8.09	−8.89
	BasketballDrive	−3.61	−3.11	−5.27	−5.05	−4.31	−5.66
	Average	−3.20	−2.58	−4.23	−4.82	−3.92	−4.79
C	RaceHorses	−4.80	−5.60	−7.62	−7.60	−8.21	−9.07
	BQMall	−3.28	−3.53	−4.96	−5.09	−5.26	−5.58
	PartyScene	−6.51	−7.45	−9.89	−8.22	−9.19	−10.75
	BasketballDrill	−4.70	−4.86	−6.58	−7.46	−7.66	−7.86
	Average	−4.82	−5.36	−7.26	−7.09	−7.58	−8.31
D	RaceHorses	−0.63	−3.00	−5.71	−2.43	−5.67	−6.91
	BQSquare	−2.81	−9.24	−10.12	−6.25	−14.24	−12.30
	BlowingBubbles	−0.28	−6.16	−9.39	−1.33	−7.74	−9.87
	BasketballPass	−1.74	−4.25	−5.39	−3.65	−7.07	−6.84
	Average	−1.36	−5.66	−7.65	−3.41	−8.68	−8.98
E	FourPeople	−1.54	−1.27	−1.81	−2.75	−2.25	−1.98
	Johnny	−1.65	−1.00	−1.87	−2.98	−2.25	−1.85
	KristenAndSara	−2.15	−1.88	−2.26	−4.42	−3.87	−2.98
	Average	−1.78	−1.39	−1.98	−3.38	−2.79	−2.27
F	BasketballDrillText	−4.74	−4.89	−5.97	−7.88	−8.08	−7.64
	ChinaSpeed	−6.25	−5.41	−5.34	−9.94	−8.84	−7.26
	SlideEditing	−1.85	−1.57	−1.51	−3.51	−3.08	−2.89
	SlideShow	−5.45	−4.88	−3.84	−8.78	−7.93	−5.32
	Average	−4.57	−4.19	−4.17	−7.52	−6.98	−5.78
	Class average	−2.80	−3.33	−4.48	−4.71	−5.27	−5.30

The highest BD-rate gains were achieved for medium- and low-resolution video test sequences (i.e., classes C and D), with average gains ranging from −3.41% to −12.82%, depending on the metric and base configuration used.

The lowest gains were obtained for class A and class E, obtaining a BD-rate gains between −1.65% and −4.01% on average.

As expected, the perceptual performance obtained by our contrast and texture masking proposals was highly dependent on the sequence type and its content but, on average, BD-rate savings of more than 5% were obtained, with particular cases achieving up to 22.89%.

As an example, the behavior of the first frame of the BQSquare sequence for all of the intra-configurations was analyzed. In Figure 12, we show the R/D curves for the first frame. As can be seen, our proposal improved the perceptual performance of the reconstructed frame for all of the metrics used. The contrast and texture masking scheme (yellow line) had the highest performance.

It is also worth noting that our proposal achieved the highest bit-rate savings at low compression rates, as can be seen in Figure 12, where for a QP of 22, we had a bit rate of 9.45 Mbps for default coding, 8.21 Mbps when contrast masking was used, and 7.72 Mbps when contrast and texture masking were used; in other words, a bit-rate saving of 18.3%.

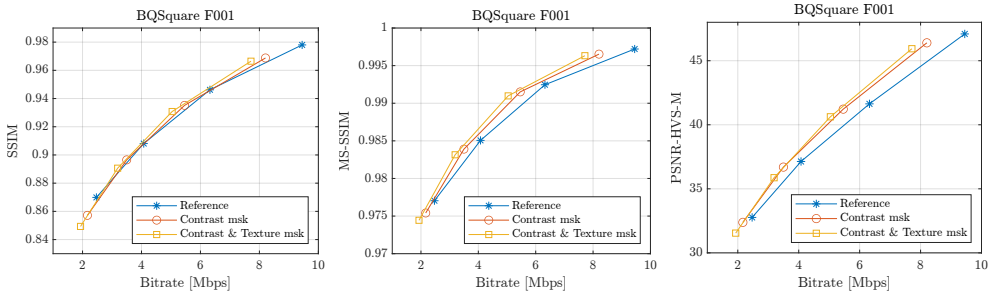
For perceptual quality, Figure 13 compares the first frame of the BQSquare sequence encoded with $QP = 22$, whose bit-rate savings we analyzed in the previous paragraph. In this case, we compared the result of the default encoding (Figure 13a) versus the en-

coding using our proposed contrast and texture masking (Figure 13b). After performing a subjective analysis, it was quite difficult to see any difference between the two pictures.

In terms of rate distortion, our proposal managed to save a considerable number of bits at the cost of a very low perceptual quality reduction.

Table 7. Average coding performance in the random-access configuration [% BD rate].

Class	Sequence Name	Constrast Masking			Contrast and Texture Masking		
		SSIM	MS-SSIM	PSNR-HVS-M	SSIM	MS-SSIM	PSNR-HVS-M
A	Traffic	−1.60	−1.30	−2.41	−4.12	−3.87	−4.07
	PeopleOnStreet	−0.98	−0.81	−1.30	−6.38	−5.95	−4.36
	Nebuta	−2.16	−1.19	−1.55	−3.53	−2.17	−1.05
	SteamLocomotiveTrain	−0.92	−0.74	−0.93	−0.79	−0.63	−0.51
	Average	−1.42	−1.01	−1.55	−3.71	−3.15	−2.50
B	Kimono	−0.39	−0.30	−0.64	−0.75	−0.60	−0.62
	ParkScene	−2.72	−1.86	−3.30	−5.02	−4.11	−4.68
	Cactus	−3.19	−2.60	−4.75	−5.52	−4.65	−5.84
	BQTerrace	−12.00	−10.32	−12.82	−15.89	−13.59	−14.28
	BasketballDrive	−3.21	−3.20	−5.33	−6.15	−5.91	−6.59
	Average	−4.30	−3.66	−5.37	−6.67	−5.77	−6.40
C	RaceHorses	−4.48	−4.89	−6.88	−8.66	−9.00	−9.39
	BQMall	−3.31	−3.37	−4.98	−6.71	−6.76	−7.13
	PartyScene	−5.67	−5.87	−9.10	−8.56	−8.67	−10.54
	BasketballDrill	−1.61	−1.90	−3.84	−5.80	−6.01	−6.00
	Average	−3.77	−4.01	−6.20	−7.43	−7.61	−8.26
D	RaceHorses	0.60	−2.45	−4.38	−4.16	−7.38	−7.39
	BQSquare	−1.57	−8.85	−10.49	−6.29	−14.72	−13.04
	BlowingBubbles	2.21	−5.30	−9.32	−0.36	−8.32	−10.83
	BasketballPass	−1.15	−3.49	−4.60	−5.67	−8.19	−7.30
	Average	0.02	−5.02	−7.20	−4.12	−9.65	−9.64
E	FourPeople	−1.44	−1.07	−1.80	−3.33	−2.75	−2.82
	Johnny	−1.90	−1.25	−2.11	−3.72	−2.81	−2.74
	KristenAndSara	−2.37	−2.06	−2.52	−4.98	−4.42	−3.84
	Average	−1.90	−1.46	−2.15	−4.01	−3.32	−3.13
F	BasketballDrillText	−1.83	−2.15	−3.65	−6.26	−6.43	−5.90
	ChinaSpeed	−6.52	−5.88	−5.40	−11.12	−10.31	−8.08
	SlideEditing	−1.30	−0.86	−2.09	−2.19	−2.19	−3.66
	SlideShow	−4.93	−4.35	−3.89	−9.72	−8.82	−6.69
	Average	−3.64	−3.31	−3.76	−7.32	−6.94	−6.08
	Class average	−2.50	−3.08	−4.37	−5.54	−6.08	−6.00



(**a**) (**b**) (**c**)

Figure 12. Rate-distortion curves of the first frame of the BQSquare sequence, comparing our proposed contrast masking (red line) and contrast and texture masking (yellow line) with the HM reference coding (blue line), using the (**a**) SSIM, (**b**) MS-SSIM, and (**c**) PSNR-HVS-M perceptual metrics.

Table 8. Average coding performance in the low-delay configuration [% BD rate].

Class	Sequence Name	Constrast Masking			Contrast and Texture Masking		
		SSIM	**MS-SSIM**	**PSNR-HVS-M**	**SSIM**	**MS-SSIM**	**PSNR-HVS-M**
A	Traffic	−1.37	−1.13	−2.40	−5.03	−4.85	−4.92
	PeopleOnStreet	−0.66	−0.72	−1.24	−6.07	−5.93	−4.33
	Nebuta	−2.29	−1.20	−1.52	−2.52	−1.37	−0.90
	SteamLocomotiveTrain	−0.71	−0.56	−0.83	−0.44	−0.07	−0.11
	Average	−1.26	−0.90	−1.50	−3.51	−3.05	−2.56
B	Kimono	−0.21	−0.16	−0.32	−0.03	0.05	0.02
	ParkScene	−1.93	−1.55	−2.67	−3.99	−3.63	−4.06
	Cactus	−2.11	−1.59	−3.68	−4.39	−3.61	−4.79
	BQTerrace	−10.42	−8.93	−13.03	−16.13	−14.36	−16.37
	BasketballDrive	−3.11	−3.08	−4.92	−6.27	−6.00	−6.52
	Average	−3.56	−3.06	−4.92	−6.16	−5.51	−6.34
C	RaceHorses	−4.27	−4.67	−7.05	−8.42	−8.82	−9.39
	BQMall	−3.36	−3.48	−5.02	−7.93	−8.01	−7.94
	PartyScene	−7.37	−7.40	−10.70	−11.57	−11.60	−13.24
	BasketballDrill	−1.13	−1.33	−2.76	−5.51	−5.69	−5.38
	Average	−4.03	−4.22	−6.38	−8.35	−8.53	−8.99
D	RaceHorses	−0.31	−2.21	−4.13	−4.99	−7.58	−6.99
	BQSquare	−8.30	−14.38	−15.77	−15.26	−22.89	−20.48
	BlowingBubbles	−2.97	−7.26	−10.74	−6.55	−11.54	−13.17
	BasketballPass	−2.64	−4.31	−5.53	−7.49	−9.55	−8.75
	Average	−3.56	−7.04	−9.04	−8.57	−12.89	−12.35
E	FourPeople	−0.20	0.01	−0.79	−2.03	−1.54	−1.20
	Johnny	−0.71	−0.35	−1.24	−4.01	−3.38	−2.99
	KristenAndSara	−1.22	−0.88	−1.45	−2.82	−2.40	−1.60
	Average	−0.71	−0.41	−1.16	−2.95	−2.44	−1.93
F	BasketballDrillText	−1.31	−1.52	−2.66	−6.28	−6.41	−5.46
	ChinaSpeed	−6.25	−5.73	−5.36	−10.81	−10.10	−7.54
	SlideEditing	−1.35	−1.48	−0.72	−3.91	−3.45	−1.99
	SlideShow	−5.59	−5.34	−5.05	−10.28	−9.75	−7.92
	Average	−3.62	−3.52	−3.45	−7.82	−7.43	−5.73
	Class average	−2.79	−3.19	−4.41	−6.23	−6.64	−6.32

(**a**) HM reference coding (9.45 Mbps)

Figure 13. *Cont*.

(**b**) Contrast and texture masking coding (7.72 Mbps)

Figure 13. Visual comparison of the first frame of the BQSquare sequence encoded at $QP = 22$. (**a**) HM reference-encoded frame; (**b**) frame encoded with contrast and texture masking.

5. Conclusions and Future Work

Compression techniques based on the HVS (e.g., texture and contrast masking) have been used for years, which proves that they are mechanisms capable of reducing the rate without impairing the image quality. In this work, we developed a novel scheme by efficiently combining contrast and texture masking techniques for the HEVC reference software showing the ability to reduce the bit rate while maintaining similar perceptual quality. We proved that by adding our proposed non-uniform 4×4 quantization matrix, we obtained an average BD-rate reduction for all of the video test sequence and the three coding modes that ranged from 2.69% (SSIM) to 4.42% (PSNR-HVS-M).

We also developed a new block classification algorithm using the mean directional variance of the image blocks and a supported vector machine, which led to a texture masking model that, in combination with contrast masking, achieved an overall average BD-rate reduction between 5.49% (SSIM) and 5.99% (MS-SSIM).

In our future work, we will (a) study the inclusion of texture over-quantization for 4×4 blocks in the HEVC reference software to further improve the RD performance of our texture masking model; (b) develop a pre-processing stage to determine when masking should not be applied at the frame level because there are sequences that hardly receive any perceptual benefit from it; and (c) evaluate other perceptual coding techniques, such as the luminance masking or the use of attention and focus metrics, which in combination with the techniques presented in this study could be able to outperform the perceptual RD performance of the HEVC reference software.

Author Contributions: Funding acquisition, O.L.-G. and G.F.E.; investigation, J.R.A., D.R.C., M.M.-R., G.V.W. and M.P.M.; software, J.R.A., D.R.C., G.F.E. and M.M.-R.; supervision, M.P.M., O.L.-G. and G.V.W.; validation, O.L.-G. and M.P.M.; writing—original draft, J.R.A., M.P.M., G.V.W., M.M.-R. and O.L.-G.; writing—review and editing, M.P.M., G.F.E., G.V.W. and O.L.-G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by MCIN/AEI/10.13039/501100011033 and by "ERDF A way of making Europe" under grants PID2021-123627OB-C55 and PID2021-123627OB-C52. This research was also funded by the Valencian Ministry of Innovation, Universities, Science and Digital Society (Generalitat Valenciana) under grants CIAICO/2021/278 and GV/2021/152 as well as by Junta de Comunidades de Castilla-La Mancha under grant SBPLY/21/180501/000195.

Data Availability Statement: We value reproducibility in research. Though our application is still in development, we can provide the current source code upon request. Interested researchers should contact the corresponding author(s) listed in the manuscript.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Video Sequence Screenshots

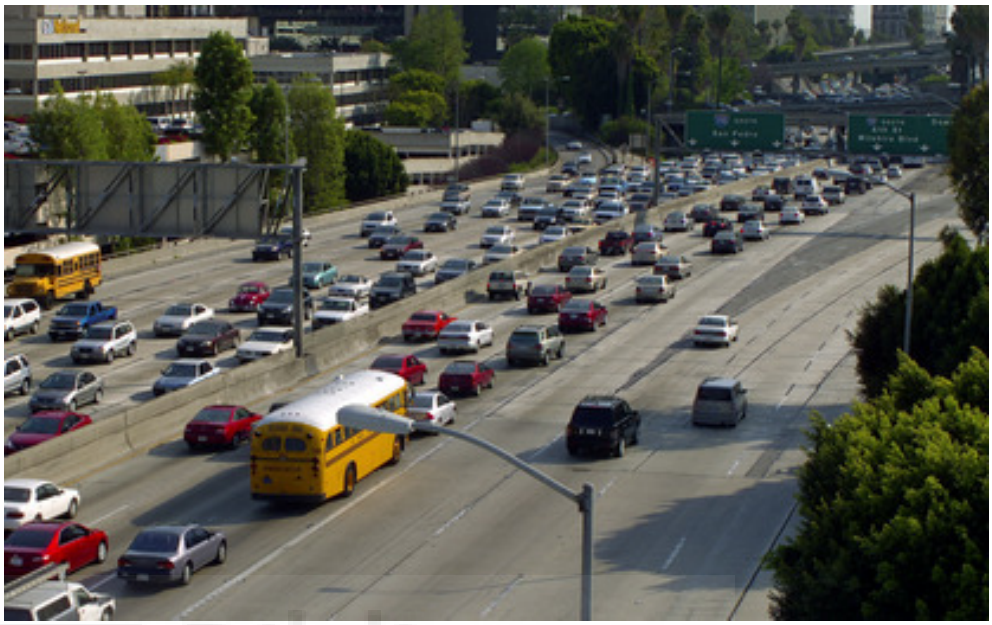


Figure A1. Traffic 2560 × 1600 30 fps Class A.

Figure A2. PeopleOnStreet 2560 × 1600 30 fps Class A.

Figure A3. NebutaFestival 2560 × 1600 60 fps Class A.

Figure A4. SteamLocomotiveTrain 2560 × 1600 60 fps Class A.

Figure A5. Kimono 1920 × 1080 24 fps Class B.

Figure A6. ParkScene 1920 × 1080 24 fps Class B.

Figure A7. Cactus 1920 × 1080 50 fps Class B.

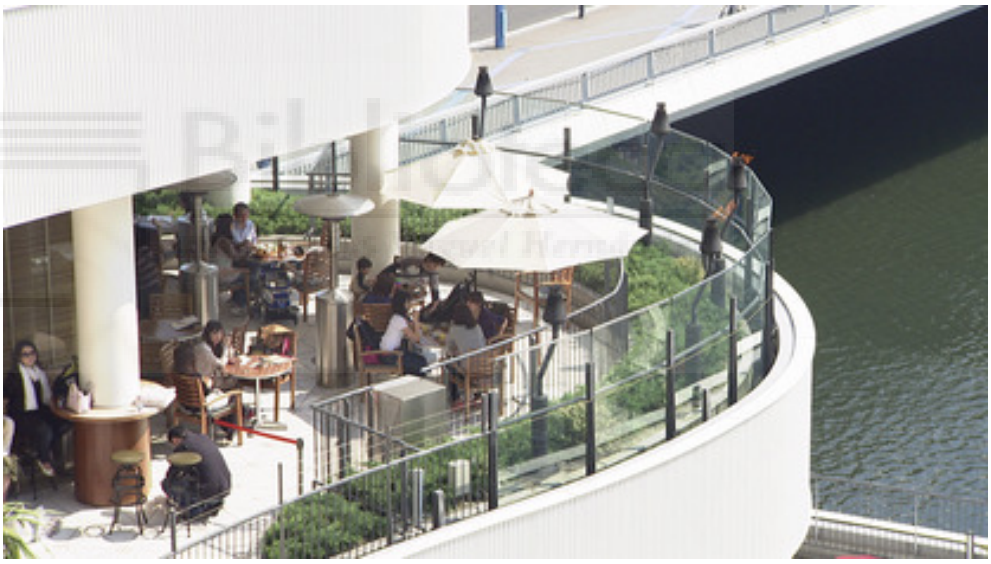


Figure A8. BQTerrace 1920 × 1080 60 fps Class B.

Figure A9. BasketballDrive 1920 × 1080 50 fps Class B.

Figure A10. RaceHorses 832 × 480 30 fps Class C.

Figure A11. BQMall 832 × 480 60 fps Class C.

Figure A12. PartyScene 832 × 480 50 fps Class C.

Figure A13. BasketballDrill 832 × 480 50 fps Class C.

Figure A14. RaceHorses 416 × 240 30 fps Class D.

Figure A15. BQSquare 416 × 240 60 fps Class D.

Figure A16. BlowingBubbles 416 × 240 50 fps Class D.

Figure A17. BasketballPass 416 × 240 50 fps Class D.

Figure A18. FourPeople 1280 × 720 60 fps Class E.

Figure A19. Johnny 1280 × 720 60 fps Class E.

Figure A20. KristenAndSara 1280 × 720 60 fps Class E.

Figure A21. BasketballDrillText 832 × 480 50 fps Class F.

Figure A22. ChinaSpeed 1024 × 768 30 fps Class F.

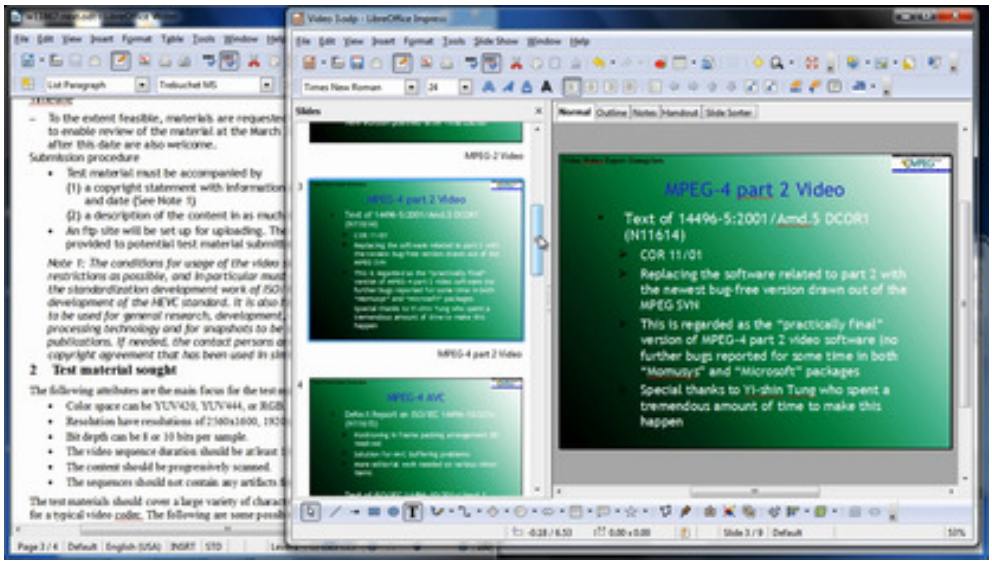


Figure A23. SlideEditing 1280 × 720 30 fps Class F.

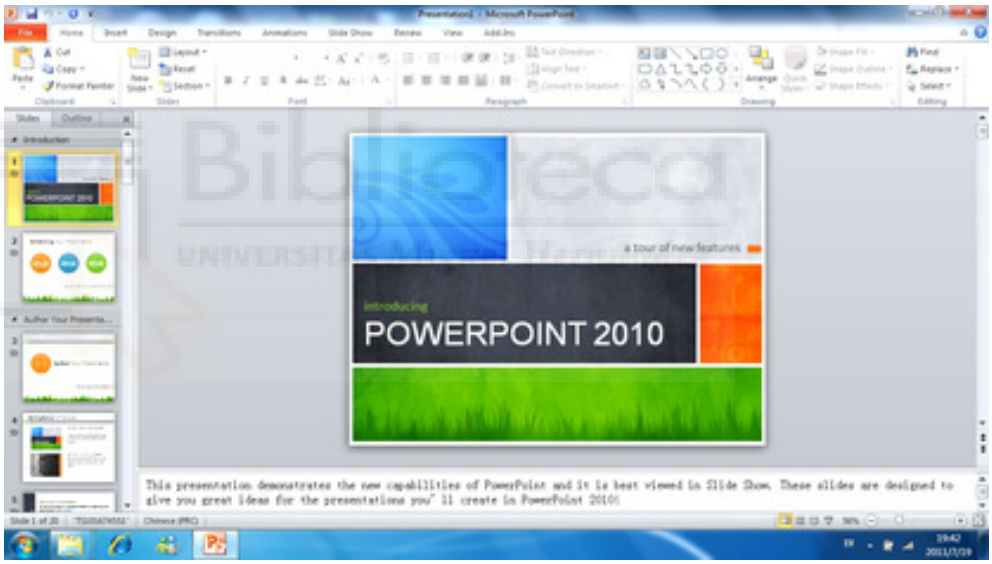


Figure A24. SlideShow 1280 × 720 20 fps Class F.

References

1. Gao, X.; Lu, W.; Tao, D.; Li, X. Image quality assessment and human visual system. In Proceedings of the Visual Communications and Image Processing 2010, Huangshan, China, 11–14 July 2010; International Society for Optics and Photonics, SPIE: San Francisco, CA, USA, 2010; Volume 7744, pp. 316–325. [CrossRef]
2. Mannos, J.; Sakrison, D. The effects of a visual fidelity criterion of the encoding of images. *IEEE Trans. Inf. Theory* **1974**, *20*, 525–536. [CrossRef]
3. Nill, N. A visual model weighted cosine transform for image compression and quality assessment. *IEEE Trans. Commun.* **1985**, *33*, 551–557. [CrossRef]
4. Daly, S. *Subroutine for the Generation of a Two Dimensional Human Visual Contrast Sensitivity Function*; Technical Report Y, 233203; Eastman Kodak: Rochester, NY, USA, 1987.
5. Ngan, K.N.; Leong, K.S.; Singh, H. Adaptive cosine transform coding of images in perceptual domain. *IEEE Trans. Acoust. Speech Signal Process.* **1989**, *37*, 1743–1750. [CrossRef]
6. Chitprasert, B.; Rao, K.R. Human visual weighted progressive image transmission. *IEEE Trans. Commun.* **1990**, *38*, 1040–1044. [CrossRef]
7. Tong, H.; Venetsanopoulos, A. A perceptual model for JPEG applications based on block classification, texture masking, and luminance masking. In Proceedings of the 1998 International Conference on Image Processing, Chicago, IL, USA, 7 October 1998; ICIP98 (Cat. No.98CB36269); Volume 3, pp. 428–432. [CrossRef]

8. *ISO/IEC 10918-1/ITU-T Recommendation T.81*; Digital Compression and Coding of Continuous-Tone Still Image. ISO: Geneva, Switzerland, 1992.
9. Zhang, X.; Lin, W.; Xue, P. Improved estimation for just-noticeable visual distortion. *Signal Process.* **2005**, *85*, 795–808. [CrossRef]
10. Wei, Z.; Ngan, K.N. Spatio-temporal just noticeable distortion profile for grey scale image/video in DCT domain. *IEEE Trans. Circuits Syst. Video Technol.* **2009**, *19*, 337–346. [CrossRef]
11. Wang, Y.; Zhang, C.; Kaithaapuzha, S. Visual Masking Model Implementation for Images & Video. In *EE368 Spring Final Paper 2009/2010*; Stanford University: Stanford, CA, USA, 2010.
12. Ma, L.; Ngan, K.N. Adaptive block-size transform based just-noticeable difference profile for videos. In Proceedings of the 2010 IEEE International Symposium on Circuits and Systems, Paris, France, 30 May–2 June 2010; pp. 4213–4216. [CrossRef]
13. Othman, Z.; Abdullah, A. An adaptive threshold based on multiple resolution levels for canny edge detection. In Proceedings of the 2nd International Conference of Reliable Information and Communication Technology (IRICT 2017), Johor, Malaysia, 23–24 April 2017; pp. 316–323. [CrossRef]
14. Gong, X.; Lu, H. Towards fast and robust watermarking scheme for H.264 Video. In Proceedings of the 2008 Tenth IEEE International Symposium on Multimedia, Berkeley, CA, USA, 15–17 December 2008; pp. 649–653. [CrossRef]
15. Mak, C.; Ngan, K.N. Enhancing compression rate by just-noticeable distortion model for H.264/AVC. In Proceedings of the 2009 IEEE International Symposium on Circuits and Systems, Taipei, Taiwan, 24–27 May 2009; pp. 609–612. [CrossRef]
16. MPEG Test Model Editing Committee. MPEG-2 Test Model 5. In Proceedings of the Sydney MPEG Meeting, Sydney, Australia, 29 March–2 April 1993.
17. Tang, C.W.; Chen, C.H.; Yu, Y.H.; Tsai, C.J. Visual sensitivity guided bit allocation for video coding. *IEEE Trans. Multimed.* **2006**, *8*, 11–18. [CrossRef]
18. McCann, K.; Rosewarne, C.; Bross, B.; Naccari, M.; Sharman, K. High Efficiency Video Coding (HEVC) Test Model 16 (HM 16) Encoder Description. In Proceedings of the 18th Meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Sapporo, Japan, 30 June–7 July 2014.
19. Prangnell, L.; Hernández-Cabronero, M.; Sanchez, V. Coding block-level perceptual video coding for 4:4:4 data in HEVC. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 2488–2492. [CrossRef]
20. Kim, J.; Bae, S.H.; Kim, M. An HEVC-compliant perceptual video coding scheme based on JND models for variable block-sized transform kernels. *IEEE Trans. Circuits Syst. Video Technol.* **2015**, *25*, 1786–1800. [CrossRef]
21. Wang, M.; Ngan, K.N.; Li, H.; Zeng, H. Improved block level adaptive quantization for high efficiency video coding. In Proceedings of the 2015 IEEE International Symposium on Circuits and Systems (ISCAS), Lisbon, Portugal, 24–27 May 2015; pp. 509–512. [CrossRef]
22. Xiang, G.; Jia, H.; Yang, M.; Liu, J.; Zhu, C.; Li, Y.; Xie, X. An improved adaptive quantization method based on perceptual CU early splitting for HEVC. In Proceedings of the 2017 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 8–10 January 2017; pp. 362–365. [CrossRef]
23. Zhang, F.; Bull, D.R. HEVC enhancement using content-based local QP selection. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 4215–4219. [CrossRef]
24. Marzuki, I.; Sim, D. Perceptual adaptive quantization parameter selection using deep convolutional features for HEVC encoder. *IEEE Access* **2020**, *8*, 37052–37065. [CrossRef]
25. Bosse, S.; Dietzel, M.; Becker, S.; Helmrich, C.R.; Siekmann, M.; Schwarz, H.; Marpe, D.; Wiegand, T. Neural Network Guided Perceptually Optimized Bit-Allocation for Block-Based Image and Video Compression. In Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 22–25 September 2019; pp. 126–130. [CrossRef]
26. Sanagavarapu, K.S.; Pullakandam, M. Object Tracking Based Surgical Incision Region Encoding using Scalable High Efficiency Video Coding for Surgical Telementoring Applications. *Radioengineering* **2022**, *31*, 231–242. [CrossRef]
27. Girod, B. What's Wrong with Mean-Squared Error? In *Digital Images and Human Vision*; MIT Press: Cambridge, MA, USA, 1993; pp. 207–220.
28. Eskicioglu, A.M.; Fisher, P.S. Image quality measures and their performance. *IEEE Trans. Commun.* **1995**, *43*, 2959–2965. [CrossRef]
29. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef] [PubMed]
30. Wang, Z.; Simoncelli, E.P.; Bovik, A.C. Multiscale structural similarity for image quality assessment. In Proceedings of the Thrity-Seventh Asilomar Conference on Signals, Systems Computers, Pacific Grove, CA, USA, 9–12 November 2003; Volume 2, pp. 1398–1402. [CrossRef]
31. Ponomarenko, N.; Silvestri, F.; Egiazarian, K.; Carli, M.; Astola, J.; Lukin, V. On between-coefficient contrast masking of DCT basis functions. In Proceedings of the Third International Workshop on Video Processing and Quality Metrics, Scottsdale, AZ, USA, 25–26 January 2007; Volume 4.
32. Martínez-Rach, M.O. Perceptual Image Coding for Wavelet Based Encoders. Ph.D. Thesis, Universidad Miguel Hernández de Elche, Elche, Spain, 2014.
33. Bjontegaard, G. Calculation of average PSNR differences between RD-Curves. In Proceedings of the ITU-T Video Coding Experts Group—Thirteenth Meeting, Austin, TX, USA, 2–4 April 2001.

34. Haque, M.; Tabatabai, A.; Morigami, Y. HVS model based default quantization matrices. In Proceedings of the 7th Meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Geneva, Switzerland, 21–30 November 2011.
35. Fraunhofer Institute for Telecommunications. HM Reference Software Version 16.20. 2018. Available online: https://vcgit.hhi.fraunhofer.de/jvet/HM/-/tags/HM-16.20 (accessed on 16 August 2024).
36. Bossen, F. Common test conditions and software reference. In Proceedings of the 11th Meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Shanghai, China, 10–19 October 2012.
37. Atencia, J.R.; Granado, O.L.; Malumbres, M.P.; Martínez-Rach, M.O.; Van Wallendael, G. Analysis of the perceptual quality performance of different HEVC coding tools. *IEEE Access* **2021**, *9*, 37510–37522. [CrossRef]
38. Ruiz-Coll, D.; Fernández-Escribano, G.; Martínez, J.L.; Cuenca, P. Fast intra mode decision algorithm based on texture orientation detection in HEVC. *Signal Process. Image Commun.* **2016**, *44*, 12–28. [CrossRef]
39. Kundu, D.; Evans, B.L. Full-reference visual quality assessment for synthetic images: A subjective study. In Proceedings of the 2015 IEEE International Conference on Image Processing (ICIP), Quebec City, QC, Canada, 27–30 September 2015; pp. 2374–2378. [CrossRef]
40. University of Southern California, Signal and Image Processing Institute. The USC-SIPI Image Database. Available online: https://sipi.usc.edu/database/ (accessed on 5 August 2024).
41. Asuni, N.; Giachetti, A. TESTIMAGES: A large-scale archive for testing visual devices and basic image processing algorithms. In Proceedings of the Smart Tools and Apps for Graphics—Eurographics Italian Chapter Conference, Cagliari, Italy, 22–23 September 2014; Giachetti, A., Ed.; The Eurographics Association: Eindhoven, The Netherlands, 2014. [CrossRef]
42. Kodak. The Kodak Color Image Dataset. Available online: https://r0k.us/graphics/kodak/ (accessed on 16 August 2024).
43. Sze, V.; Budagavi, M.; Sullivan, G.J. *High Efficiency Video Coding (HEVC): Algorithms and Architectures*; Integrated Circuits and Systems; Springer: Berlin/Heidelberg, Germany, 2014.