

Análisis combinado de texture y contrast masking en HEVC

Javier Ruiz Atencia, Otoniel López Granado, Manuel Pérez Malumbres,
Miguel O. Martínez-Rach¹

Resumen— En este artículo se han analizado dos técnicas de codificación perceptual, como son texture masking y contrast masking, aplicadas al estándar de codificación de vídeo HEVC. Para el texture masking se han adaptado algoritmos utilizados en compresión de imágenes a las características del HEVC, como es su transformada y los diferentes tamaños de bloque disponibles. Por su parte para el contrast masking se ha determinado la matriz de pesos óptima para cada tamaño de bloque. Por último se han codificado diferentes secuencias utilizando estas técnicas, obteniendo los valores BD-Rate con el objetivo de comprobar la viabilidad de su uso de forma individual y combinado.

Palabras clave— HEVC, perceptual, texture masking, contrast masking, DCT, CSF.

I. INTRODUCCIÓN

LOS codificadores de imagen y video eliminan la redundancia de información existente en las imágenes naturales utilizando técnicas de transformación al dominio de la frecuencia como la transformada discreta del coseno (DCT) del seno (DST) o wavelets (DWT), entre otras, que permiten compactar la información presente en la imagen en un conjunto reducido de coeficientes frecuenciales que serán codificados de distintas formas para reducir la cantidad de información a transmitir o almacenar. Así mismo, utilizan la cuantización como técnica para reducir el peso de dichos coeficientes, de tal manera que solo se codifican aquellos que son más importantes para la posterior reconstrucción de la imagen. El problema siempre ha sido determinar qué coeficientes proteger y no cuantizar por ser los más importantes desde el punto de vista subjetivo. La mayoría de los estándares de codificación utilizan un particionado en bloques de distintos tamaños para aplicar estas etapas de transformación y cuantización de manera repetitiva para toda la imagen.

En la codificación de imagen y video se han incorporado muchas técnicas, basadas en la percepción de la calidad de nuestro sistema visual humano (HVS). Estas técnicas tienen como objetivo mejorar la percepción subjetiva de la calidad de la imagen o el video reconstruido. La calidad de la reconstrucción se reduce como consecuencia de la etapa de cuantización, donde se produce la pérdida de información de la imagen, apareciendo artefactos o distorsiones visibles por nuestro HVS que no estaban presentes en la imagen original.

Sin embargo el HVS no es capaz de detectar, bajo ciertas circunstancias, dichas distorsiones cuando

están enmascaradas por una alta textura, bajos o altos contrastes, baja o alta luminosidad, etc. Por tanto las condiciones del HVS que producen este enmascaramiento de los defectos han sido ampliamente estudiadas para ofrecer mecanismos que permitan eliminar selectivamente más información en aquellas zonas de la imagen donde los errores en la reconstrucción van a ser enmascarados.

Una de las propiedades más utilizada es la incorporación, normalmente en misma etapa de cuantización, de la Contrast Sensitivity Function (CSF) que pone de manifiesto que determinadas variaciones de contraste entre un objeto y su fondo no son percibidas por el sistema visual humano (HVS), siendo éstas dependientes de la luminancia, la distancia de visualización, entre otros factores. Otro factor de enmascaramiento lo produce la luminancia y se conoce como Luminance Masking ya que en zonas con mucha o muy poca luminancia estos errores pasan más desapercibidos. También la presencia de textura en la imagen proporciona enmascaramiento de los errores en la reconstrucción, así un error en una superficie homogénea es más detectable que en una zona con mucha textura.

En [1] Tong propuso un modelo perceptual del enmascaramiento por textura y luminancia proponiendo a su vez un método para la clasificación de los bloques de coeficientes DCT en función del tipo de contenido, textura, borde o plano. Para realizar esta clasificación utiliza un algoritmo que se basa en el peso de los coeficientes agrupados en función de su frecuencia o posición dentro del bloque. Con el modelo de enmascaramiento y la clasificación de los bloques se determina el grado de cuantización adicional aplicable a un bloque para que las distorsiones producidas por la cuantización sigan enmascaradas.

El modelo de [1] ha sido refinado y variado por otros autores, por ejemplo en [2] Zhang, modifica el modelo de luminance maskig y el clasificador utilizando una ponderación del peso de los coeficientes del bloque en vez de usar la suma absoluta. Esto modifica el algoritmo de clasificación añadiendo también el efecto de intra-band masking. Este efecto se refiere a la tolerancia de error imperceptible dentro de la propia subbanda. Es decir, para cada coeficiente dentro del bloque de 8×8 se aplica un valor diferente de cuantización en función de la clasificación del bloque y de la posición del coeficiente.

Como se indica en [3] la mayoría de los modelos se basan en particionados de la imagen en bloques de 8×8 [2][4][5]. En [3] los autores extienden la clasificación a bloques de 16×16 para adaptarse a las

¹Dpto. de Ingeniería de Computadores, Universidad Miguel Hernández de Elche, e-mail: mmrach@umh.es.

resoluciones de la imagen soportadas por los últimos estándares, sin embargo utilizan un modelo de clasificación en el dominio del píxel basado en el algoritmo de detección de bordes Canny, aplicando una cuantización adaptativa en función del tamaño del bloque. También son varios los autores [6][7] que utilizan una reducción a 4×4 del clasificador basado en [1].

Otro aspecto a considerar a la hora de incluir texture masking en un codificador, es la señalización en el bitstream del tipo de bloque o bien del valor de cuantización adicional a aplicar en cada bloque. La mayoría de los autores citados utiliza los umbrales definidos por el modelo JND para descartar completamente los coeficientes por debajo del umbral, de forma compatible con el estándar correspondiente, sin necesidad por tanto de enviar información adicional al decodificador.

Todos estos trabajos utilizan el PSNR como métrica de calidad al evaluar el rendimiento R/D. Pero como es sabido el PSNR no refleja fielmente la valoración perceptual de la calidad. Por eso en algunos estudios como en [3] se han realizado test subjetivos para validar su modelo JND utilizando el Difference Mean Opinion Score (DMOS) como indicador de la calidad perceptual.

En este trabajo nos hemos centrado en texture y contrast masking y su adaptación para el codificador HEVC. Utilizando como base el algoritmo de [2], se ha adaptado para la inclusión del texture masking en el codificador de video HEVC, modificando los umbrales para todos los tamaños de bloque. Utilizamos la versión que añade información adicional (multiplicador por cada bloque) al bitstream. Para reducir el volumen que ésta ocupa se ha aplicado un algoritmo simple de agrupación de valores de masking utilizando valores estadísticos y aplicándolo adaptativamente al tamaño de bloque. Se han utilizado las matrices estándar incluidas en el HEVC (perceptual weighting matrices) para incluir el contrast masking en bloques desde 8×8 a 32×32 . El estándar no aplica la CSF para bloques de 4×4 por lo que nosotros proponemos la inclusión de una matriz perceptual específica para ese tamaño [8] con el fin de aumentar la tasa de compresión para la misma calidad perceptual.

El presente estudio utiliza la codificación en modo intra para imágenes y all intra para video y analiza el comportamiento combinado y por separado de la CSF y el texture masking para un particionado en bloques de tamaño fijo dese 4×4 hasta 32×32 . Los resultados del rendimiento R/D se presentan utilizando el PSNR como métrica de distorsión pero también un conjunto amplio de métricas de calidad objetiva, SSIM, MSSSIM, VIFP, VIF, PSNRHVS y PSNRHVSM [9][10]. Para proporcionar los resultados de ahorro en rate se utiliza el método Bjøntegaard [11] adaptado a las distintas métricas.

El resto del artículo se estructura de la siguiente manera. En el capítulo II se desarrolla ampliamente la adaptación al estándar HEVC del texture masking. En el capítulo III se define nuestra propuesta

de aplicar CSF a cada tamaño de bloque disponible. En el capítulo IV se muestran resultados de aplicar o no las técnicas de enmascaramiento para una serie de imágenes y secuencias de vídeo. Por último, en el capítulo V se resumen las conclusiones del estudio así como las líneas futuras de investigación para seguir ampliando este trabajo.

II. ADAPTACIÓN DE HEVC PARA TEXTURE MASKING

Como se ha mencionado en la introducción para este trabajo hemos utilizado el algoritmo clasificador de Zhang [2], cuyo diagrama de flujo se muestra en la Figura 1. Zhang asigna los siguientes valores a los umbrales del clasificador: $\mu_1 = 125$, $\mu_2 = 290$, $\mu_3 = 900$, $\alpha_1 = 7$, $\beta_1 = 5$, $\sigma = 16$ y $\kappa = 0,1$. Las expresiones E_1 y E_2 indican dos medidas utilizadas para determinar la presencia de bordes en un bloque mientras que $TexE$ indica su nivel de textura. Estas expresiones se obtienen mediante las Ecuaciones (1), (2) y (3).

$$E_1 = (\bar{L} + \bar{M})/\bar{H} \quad (1)$$

$$E_2 = \bar{L}/\bar{M} \quad (2)$$

$$TexE = M + H \quad (3)$$

donde M y H corresponden a la suma absoluta de los coeficientes de media frecuencia (MF) y alta frecuencia (HF) respectivamente; y \bar{L} , \bar{M} y \bar{H} corresponden al valor promedio de los coeficientes absolutos de baja (LF), media (MF) y alta frecuencia (HF) respectivamente. Los coeficientes se clasifican en función de la posición que ocupen en la matriz de coeficientes transformados de tamaño 8×8 (Figura 2).

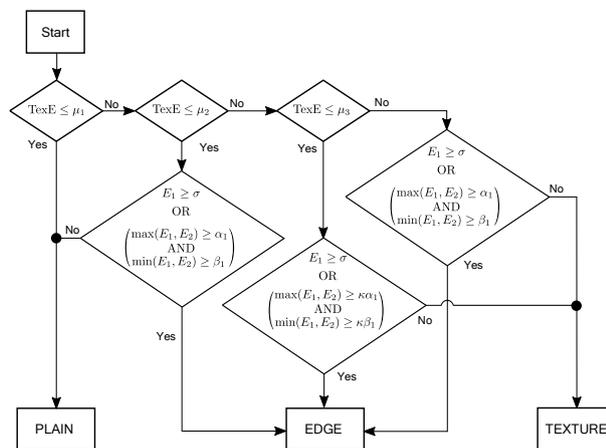


Fig. 1: Diagrama del clasificador de bloques utilizado.

Por otro lado se obtiene el factor multiplicador de masking en función de la clasificación del bloque. Para bloques TEXTURE se utiliza la Ecuación (4):

$$m(k) = (MaxElevation - 1) \frac{TexE(k) - MinE}{MaxE - MinE} + 1 \quad (4)$$

donde [1] y [2] establecen los parámetros $MaxElevation = 2,25$, que corresponde al valor

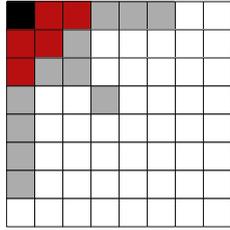


Fig. 2: Clasificación de coeficientes transformados para bloques de 8×8 . Diferenciamos LF (rojo), MF (gris) y HF (blanco) como baja, media y alta frecuencia respectivamente.

máximo que puede alcanzar el factor multiplicador de masking, y las energías mínimas $MinE = 290$ y máximas $MaxE = 1800$, obtenidos empíricamente.

Tong [1] y Zhang [2], así como buena parte de los trabajos derivados de éstos basan la clasificación en bloques de tamaño 8×8 píxeles transformados mediante la Transformada Discreta de Coseno, debido a que codifican la imagen utilizando estándares cuya transformada es la DCT-II (JPEG) o alguna aproximación finita de la misma (H.264/AVC). A diferencia de éstos estándares, HEVC utiliza tamaños de bloque desde 4×4 hasta 32×32 y una aproximación finita a enteros y escalada de la DCT-II para que los vectores base de cada tamaño de matriz de transformación tengan la misma energía. Además, Tong y Zhang trabajan con la imagen original mientras que el estándar HEVC, al igual que el resto de estándares basados en el esquema Hybrid Video Coding, trabaja con el error residual de la mejor predicción calculada. Es por todo esto que en el presente trabajo se ha estudiado la adaptación del algoritmo clasificador de bloques así como de las ecuaciones para obtener el factor multiplicador de masking con el objetivo de integrarlo en el estándar HEVC.

A. Adaptación a la transformada HEVC

Debido a los cambios introducidos en la transformada del estándar HEVC resulta necesario realizar una adaptación de los umbrales con el objeto de que algoritmo clasifique los bloques de igual forma a como lo haría utilizando la DCT-II. Para ello se ha comparado el ratio de los coeficientes de ambas transformadas aplicadas a bloques de tamaño 8×8 para un conjunto de imágenes de muestra con diferentes niveles de textura. Filtrando los valores atípicos debido a ratios entre coeficientes con energía muy baja y aplicando la media obtenemos un factor multiplicador aproximado de 16. En la Figura 3 se expone a modo de ejemplo la relación entre los coeficientes de ambas transformadas para un bloque cualquiera.

Obtenido este multiplicador se ha determinado que los umbrales a modificar son solamente μ_1 , μ_2 , μ_3 , $MinE$, $MaxE$ y δ y cuyos valores adaptados son 2000, 4640, 14400, 4640, 28800 y 6400 respectivamente. Esto es porque las expresiones que evalúa dichos umbrales, como por ejemplo $TexE$, se obtienen a partir de la suma de coeficientes. Sin embargo, el

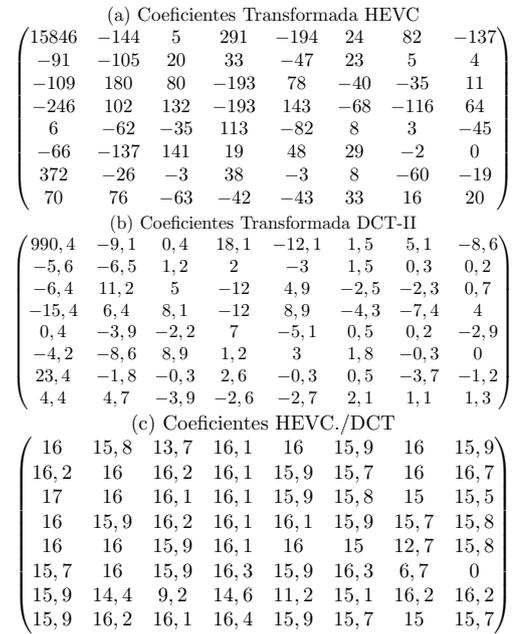


Fig. 3: Coeficientes transformados para un bloque de tamaño 8×8 : transformada del estándar HEVC (a), transformada DCT-II (b) y ratio entre ambas transformadas (c)

resto de umbrales del algoritmo clasificador son evaluados por las expresiones E_1 y E_2 , Ecuaciones (1) y (2), en donde el multiplicador no es tenido en cuenta y por tanto conservan el mismo valor dado por [2]. En la Figura 4 se muestra un ejemplo del clasificador de bloques adaptado al estándar HEVC para una imagen de test. Como se puede observar, la clasificación utilizando la transformada HEVC con los umbrales adaptados, si bien no es perfecta, es bastante fiel a la realizada mediante la DCT-II.

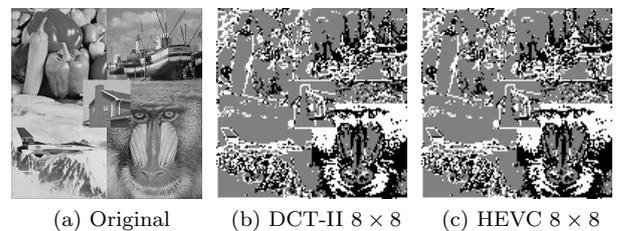


Fig. 4: Clasificación de bloques 8×8 de una imagen de test (a) utilizando la transformada DCT-II (b) y la transformada HEVC con umbrales adaptados (c). En gris los bloques PLAIN, en blanco los EDGE y en negro los TEXTURE.

B. Adaptación a tamaños de bloque

Para adaptar el texture masking a los diferentes tamaños de bloque integrados en el estándar HEVC, se ha elaborado un análisis de los coeficientes transformados con el fin de extender el modelo propuesto por Tong [1], basado en la agrupación de estos coeficientes según su nivel frecuencial, y diferenciándonos de [3], donde los autores realizan la clasificación en el dominio del píxel mediante el algoritmo de detección

de bordes Canny.

La primera parte del análisis ha consistido en construir el modelo de agrupación en niveles frecuenciales de los coeficientes transformados. Para ello se han generado transformadas de bloques aleatorios de diferentes imágenes de muestra que se han redimensionado con el objetivo de que los bloques a transformar sean visualmente idénticos para cada tamaño de bloque. Las imágenes originales se han utilizado para la transformada 8×8 ; para la transformada de tamaño 4×4 se ha reducido a la mitad la imagen original mientras que para las transformadas 16×16 y 32×32 se ha aumentado en dos y cuatro veces el tamaño de la imagen respectivamente.

A partir de este análisis se ha observado cierta correlación entre el peso que tienen los coeficientes de la transformada 8×8 y el peso de esos mismos coeficientes para las transformadas de tamaño superiores. Esto era de esperar debido a las propiedades de compactación de energía de la DCT. Sin embargo para tamaños de 4×4 resulta más complejo determinar qué coeficientes determinan los diferentes niveles frecuenciales. Es por eso que en este trabajo no se ha aplicado texture masking en bloques de este tamaño.

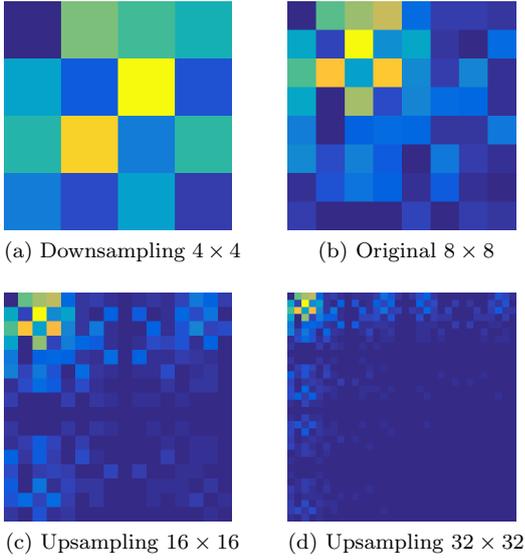


Fig. 5: Mapa de calor de los coeficientes AC absolutos normalizados de un mismo bloque reescalado usando la transformada DCT-II.

En la Figura 5 se muestra a modo de ejemplo los coeficientes transformados, representados como un mapa de calor, para cada tamaño disponible de un mismo bloque utilizando las redimensiones oportunas. Para una mejor visualización se ha anulado la componente DC. Comprobamos como el patrón que dan los pesos en la imagen original (b) se repite para los tamaños superiores (c) y (d), mientras que dicho patrón se ve alterado para bloques de tamaño 4×4 (a).

A partir de este análisis hemos definido la clasificación utilizando los 8×8 primeros coeficientes de cada bloque para los tamaños 16×16 y 32×32 .

Una vez se tiene la clasificación de los coeficientes se ha realizado otro estudio para determinar la rela-

ción entre estos mismos coeficientes para diferentes tamaños de bloque. Para ello se siguió la metodología vista en el apartado II-A, esto es, se ha utilizado un conjunto variado de imágenes de muestra para realizar el particionado 8×8 , y a su vez se han redimensionado para que visualmente el particionado en bloques de 16×16 y 32×32 sea equivalente al particionado de la imagen original. Hecho esto se ha extraído de cada bloque los valores E_1 , E_2 y $TexE$ y se han representado para ver su relación.

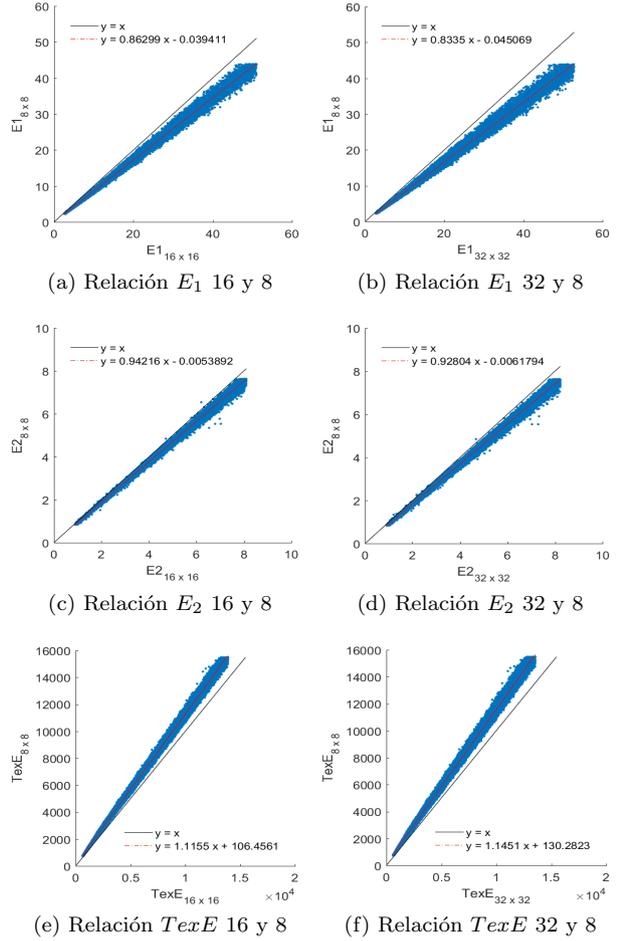


Fig. 6: Estudio estadístico de la relación entre el valor de las expresiones E_1 (a,b), E_2 (c,d) y $TexE$ (e,f) para bloques de tamaño 16×16 y 32×32 frente a bloques de tamaño 8×8

En la Figura 6 se muestra el resultado de dicho análisis. Como se puede observar existe una relación lineal entre las expresiones para tamaños de bloque 16×16 y 32×32 con respecto al tamaño 8×8 . A partir de estos datos se ha extraído la recta de regresión para cada una de las expresiones, aplicándose al algoritmo clasificador y dando como resultado una clasificación prácticamente igual a la clasificación original, tal y como se observa en la Figura 7.

Para obtener el factor multiplicativo de masking también es necesario realizar una adaptación de sus umbrales a los tamaños de bloque superiores a 8×8 . El factor multiplicativo es un valor numérico que modifica (cuantiza) los coeficientes AC de la transformada de cada bloque en función de la clasificación del

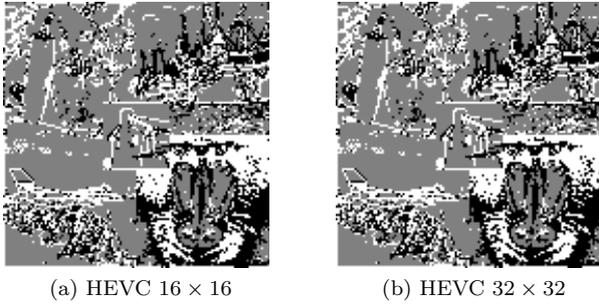


Fig. 7: Clasificación de una imagen de test utilizando la transformada HEVC con umbrales adaptados para tamaño 16×16 (a) y 32×32 (b).

mismo y de su nivel de energía de textura. Para este trabajo se ha decidido utilizar el esquema propuesto por Tong, explicado al comienzo del Apartado II.

Como ya hemos visto, los umbrales $MinE$, $MaxE$ y $\delta_{threshold}$ se han adaptado para la transformada HEVC y bloques de tamaño 8×8 , sin embargo estos valores deben ajustarse también para bloques de tamaño superior.

En la Tabla I se muestran los umbrales adaptados para obtener el valor de masking.

TABLA I: Valor de los umbrales para la obtención del factor de masking a diferentes tamaños de bloque

Variables umbral	Tamaño 8x8	Tamaño 16x16	Tamaño 32x32
$MinE$	4640	4054	3925
$MaxE$	28800	25160	24364
$\delta_{threshold}$	6400	6189	6136

C. Aplicación al residuo

Una vez se tiene el algoritmo clasificador adaptado a la transformada HEVC así como a diferentes tamaños de bloque se ha estudiado su implementación para las imágenes residuales, ya que en los apartados anteriores se ha aplicado la transformada a imágenes originales.

Primero se ha intentado buscar una relación entre las expresiones E_1 , E_2 y $TexE$ para los coeficientes transformados de imágenes originales e imágenes residuales de forma análoga al estudio realizado en el Apartado II-B. El resultado de este estudio indica que no es posible la adaptación de los coeficientes ya que hay una dispersión elevada.

Eso nos hizo plantearnos si realmente era necesario obtener la misma clasificación para los bloques residuales que si se estuvieran evaluando los bloques originales. La imagen residual tiene otros niveles de textura y un rango dinámico que varía entre -255 y 255 (para imágenes con 8 bits de profundidad), pero para el clasificador de bloques sus coeficientes transformados son como los de cualquier otra imagen natural. En la Figura 8 (a) se muestra error residual de la predicción correspondiente a la imagen de ejemplo utilizada en este artículo, mientras que en (b) observamos el resultado de la clasificación de sus bloques

transformados.

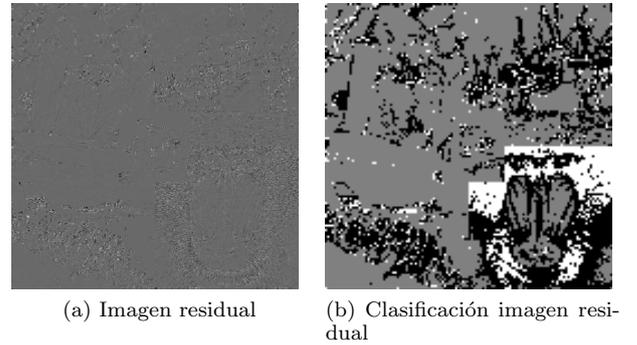


Fig. 8: Imagen residual de test para el estudio del *masking* para un tamaño de bloque de 8×8 correspondiente a la Figura 4 (a)

Si observamos con detenimiento (a) podemos apreciar cierta similitud con su imagen original: cuanto mayor nivel de textura tiene el bloque original mayor es el error de predicción, lo cual da la apariencia de textura a la imagen residual, e inversamente ocurre lo mismo para los bloques planos, cuyo residuo carece de textura debido a la facilidad de predicción usando los modos Planar y DC del estándar HEVC. Es por esto por lo que la clasificación de los bloques TEXTURE y PLAIN en el residuo es muy similar a la obtenida para la imagen original.

En cambio no ocurre lo mismo con los bloques clasificados como EDGE, ya que comprobamos como la mayoría de estos bloques han dejado de detectarse, pasando a ser en su mayoría bloques TEXTURE y en algunos casos PLAIN. Esto ocurre porque el estándar HEVC tiene 35 modos de predicción, de los cuales 33 corresponden a predicciones angulares, es decir, predicen la presencia de un borde en 33 direcciones angulares diferentes. Si se tienen bloques con bordes fuertemente marcados es de esperar que el codificador seleccione un modo de predicción angular de forma que el error residual tenga un nivel de borde muy bajo o, en caso de tener una predicción muy buena, inexistente.

Llegando a esta conclusión hemos optado por no adaptar o reajustar los umbrales o expresiones del algoritmo clasificador de bloques para obtener el mismo resultado que la clasificación en imágenes originales, ya que estamos clasificando el residuo y no la imagen original.

III. ADAPTACIÓN DE HEVC PARA CONTRAST MASKING

El contrast masking o enmascaramiento por contraste es una técnica de compresión perceptual que explota el efecto visual por el cual el ojo humano no es capaz de percibir pequeñas variaciones de luminancia cuando el fondo de la escena es lo suficientemente claro u oscuro. 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 [12][13][14], sin embargo el modelo más extendido en el ámbito de la codificación de imagen y vídeo es el de Mannos y Sakrison [15].

El estándar HEVC incorpora cuantización dependiente de la frecuencia, esto es, aplica diferente nivel de cuantización dependiendo del peso que tengan los coeficientes en el dominio frecuencial basándose para ello en el modelo del Sistema Visual Humano (HVS) [16][17], utilizando para ello matrices de cuantización no-uniformes, también llamadas matrices de pesos. Para codificación intra-luminancia se define esta matriz de cuantización únicamente para tamaños de bloque 8×8 , mientras que para tamaños superiores (16×16 y 32×32) las matrices se obtienen mediante sobre-muestreo y replicación. Para el caso de los bloques de tamaño 4×4 el estándar no define matriz de cuantización de pesos.

En este trabajo se ha introducido la matriz de cuantización de pesos para tamaños de bloque 4×4 con el objetivo de incrementar el nivel de compresión, mientras que para el resto de bloques se ha utilizado la matriz que incorpora el estándar. Para obtener la matriz de pesos 4×4 propuesta se ha empleado el estudio realizado por Martínez-Rach [8], y cuyo valor final es mostrado en la Figura 9 (b).

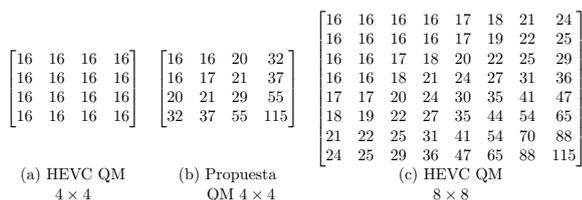


Fig. 9: Matrices de cuantización por defecto para tamaños de bloque 4×4 (a) y 8×8 (c), así como nuestra propuesta no-uniforme para tamaño de bloque 4×4 (b).

A modo de ejemplo se muestra en la Figura 10 una comparativa entre aplicar o no nuestra propuesta de matriz de pesos a bloques de tamaño 4×4 . Se ha demostrado que la métrica PSNR es poco consistente en lo referente a la percepción del HVS [18]. En la figura comparamos ambas curvas utilizando la métrica de calidad perceptual MS-SSIM [19], donde observamos que aplicando CSF se obtiene un mejor resultado.

IV. RESULTADOS

Para obtener los resultados de evaluación de las técnicas de enmascaramiento analizadas en este trabajo se ha utilizado un emulador Intra-HEVC desarrollado en Matlab por la Universidad Miguel Hernández de Elche y cuyos resultados se han validado con el software de referencia HM 14.0. Se han escogido una serie de imágenes y secuencias de vídeo con tamaños y nivel de textura variada. Para cada secuencia de vídeo se ha codificado 1 segundo de metraje, cuyo número de fotogramas varía en función de los FPS.

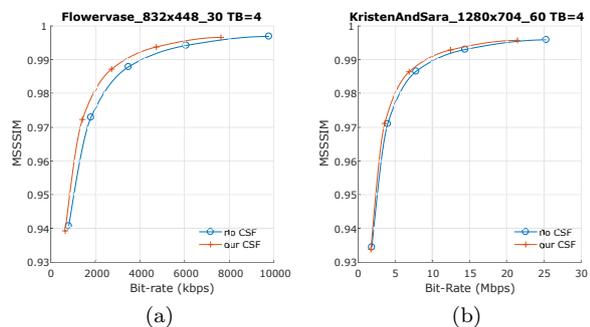
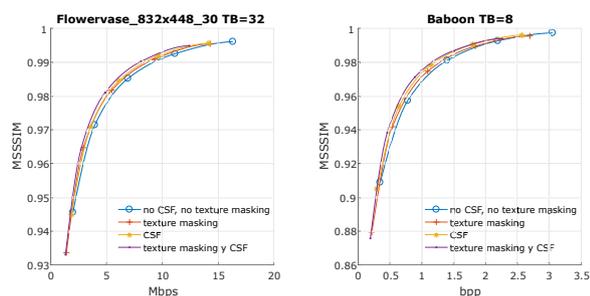
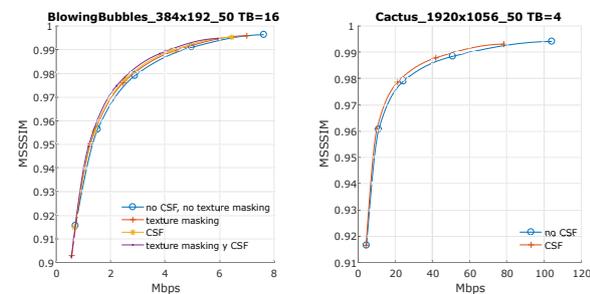


Fig. 10: Comparativa de curvas R-D entre aplicar o no nuestra propuesta de CSF para varias secuencias de vídeo particionadas en bloques fijos de tamaño 4×4 .

Las curvas Rate-Distortion se han obtenido aplicando texture y contrast masking de forma individual y conjuntamente. Para el caso de los bloques de tamaño 4×4 únicamente se ha aplicado contrast masking puesto que no se ha desarrollado el texture masking tal y como se ha explicado en el Apartado II-B. Las curvas donde se ha aplicado texture masking (individual o junto con contrast masking) incluyen el overhead de bits producto de enviar la matriz de masking por el bitstream. En la Figura 11 se muestran ejemplos de curvas rate-distortion para cada tamaño de bloque disponible.



(a) Curva MS-SSIM - Mbps para tamaño de bloque 32×32 (b) Curva MS-SSIM - bpps para tamaño de bloque 16×16



(c) Curva MS-SSIM - Mbps para tamaño de bloque 8×8 (d) Curva MS-SSIM - Mbps para tamaño de bloque 4×4

Fig. 11: Ejemplo de curvas Rate-Distortion para diferentes imágenes y secuencias de vídeo.

Con el fin de extraer un valor numérico para determinar la mejora o no de ahorro en rate se utiliza el cálculo Bjøntegaard-Delta Rate (BD-Rate) [11] sobre la métrica de calidad MS-SSIM para cada una de

las secuencias analizadas, donde los valores negativos indican un ahorro en rate. En las Tablas II a V se muestra el resultado del estudio realizado para cada uno de los tamaños de bloque definidos.

TABLA II: Resultados BD-Rate (%) para métrica perceptual MS-SSIM y tamaño de bloque 32×32 .

Secuencia test	Texture masking	Contrast masking	Texture y contrast masking
Baboon (imagen)	-2.06	-8.90	-10.39
Lena (imagen)	-5.25	-3.81	-9.49
BlowingBubbles	-2.03	-5.01	-6.57
FlowerVase	-7.56	-7.56	-13.92
RaceHorses	-3.08	-4.34	-7.25
KristenAndSara	-9.09	-2.41	-10.91
Kimono	-4.18	-0.50	-4.50
Cactus	-6.90	-2.55	-8.84
Promedio	-5.02	-4.39	-8.98

TABLA III: Resultados BD-Rate (%) para métrica perceptual MS-SSIM y tamaño de bloque 16×16 .

Secuencia test	Texture masking	Contrast masking	Texture y contrast masking
Baboon (imagen)	-2.15	-9.15	-10.86
Lena (imagen)	-8.60	-3.00	-10.47
BlowingBubbles	-4.80	-5.91	-9.10
FlowerVase	-10.36	-8.68	-17.06
RaceHorses	-5.35	-5.31	-9.21
KristenAndSara	-9.29	-3.08	-11.46
Kimono	-2.25	-0.67	-2.63
Cactus	-7.35	-3.17	-9.54
Promedio	-6.27	-4.87	-10.04

TABLA IV: Resultados BD-Rate (%) para métrica perceptual MS-SSIM y tamaño de bloque 8×8 .

Secuencia test	Texture masking	Contrast masking	Texture y contrast masking
Baboon (imagen)	-6.06	-11.03	-15.02
Lena (imagen)	-3.94	-5.59	-9.51
BlowingBubbles	5.47	-8.01	-10.84
FlowerVase	-10.62	-10.70	-18.64
RaceHorses	-3.75	-7.53	-8.60
KristenAndSara	-6.51	-4.48	-10.01
Kimono	-1.03	-1.18	-1.97
Cactus	-5.99	-4.87	-9.50
Promedio	-5.42	-6.67	-10.51

Podemos observar como para todos los tamaños de bloque aplicar técnicas de masking mejora el ahorro en rate con respecto a no utilizarla, aunque existen diferencias notables entre las distintas secuencias utilizadas. Esto ocurre porque se está comprimiendo la imagen en función de la escena con técnicas perceptuales, y por tanto la eficiencia de utilizar estas técnicas dependerá del nivel de textura (texture masking) y de las componentes frecuenciales de los diferentes objetos de la escena (contrast masking). Cabe destacar también que el uso de aplicar ambas técnicas de enmascaramiento vistas en este trabajo produce

TABLA V: Resultados BD-Rate (%) para métrica perceptual MS-SSIM y tamaño de bloque 4×4 .

Secuencia test	Contrast masking
Baboon (imagen)	-17.62
Lena (imagen)	-9.12
BlowingBubbles	-14.57
FlowerVase	-19.02
RaceHorses	-13.15
KristenAndSara	-9.69
Kimono	-1.97
Cactus	-9.12
Promedio	-11.78

siempre un mejor resultado que aplicar únicamente una de ellas.

En lo que respecta a utilizar CSF en bloques de tamaño 4×4 tal y como se observa en la Tabla V podemos concluir que su uso está justificado, y es donde se obtiene un mejor BD-Rate con respecto a los otros tamaños de bloque.

V. CONCLUSIONES Y TRABAJO FUTURO

Las técnicas de compresión basadas en el sistema visual humano (HVS) como son el texture y contrast masking han sido utilizadas durante años demostrando que son una herramienta capaz de reducir el rate sin que ello perjudique a la calidad de la imagen. En este trabajo se han adaptado dichas técnicas a las características propias del estándar HEVC y se ha demostrado que su uso sigue siendo válido para reducir la tasa de bits.

Como trabajos futuros se pretende (a) seguir adaptando las técnicas de masking para tamaños superiores a 32×32 para utilizarlos en los nuevos estándares, como son JEM y VVC, (b) proponer algoritmos para reducir el overhead producido por el texture masking, y (c) incluir otras técnicas de compresión perceptual, como por ejemplo el luminance masking.

En cuanto al estándar HEVC, como se ha indicado anteriormente, se ha utilizado una herramienta en Matlab cuyo particionado se determina en bloques de tamaño fijo. Queda pendiente aplicar las técnicas de masking para el particionado Quad-tree propio del estándar.

Por último, por mucho que la métrica MS-SSIM esté optimizada para comparaciones perceptuales de imágenes, queda pendiente realizar estudios subjetivos DMOS.

AGRADECIMIENTOS

Este trabajo ha sido financiado por el Ministerio de Ciencia, Innovación y Universidades con referencia RTI2018-098156-B-C54 cofinanciado con fondos FEDER (MINECO/FEDER/UE).

REFERENCIAS

- [1] H.H.Y. Tong and A.N. Venetsanopoulos, "A perceptual model for jpeg applications based on block classification, texture masking, and luminance masking," in *Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on*, Oct 1998, pp. 428–432 vol.3.

- [2] X.H. Zhang, W.S. Lin, and P. Xue, "Improved estimation for just-noticeable visual distortion," *Signal Processing*, vol. 85, no. 4, pp. 795 – 808, 2005.
- [3] L. Ma and K. N. Ngan, "Adaptive block-size transform based just-noticeable difference profile for videos," in *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, May 2010, pp. 4213–4216.
- [4] Z. Wei and 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, no. 3, pp. 337–346, March 2009.
- [5] Yuhong Wang, Chi Zhang, and Sukesh Kaithapuzha, "Visual masking model implementation for images & video," 2010.
- [6] X. Gong and H. Lu, "Towards fast and robust watermarking scheme for h.264 video," in *2008 Tenth IEEE International Symposium on Multimedia*, Dec 2008, pp. 649–653.
- [7] C. Mak and K. N. Ngan, "Enhancing compression rate by just-noticeable distortion model for h.264/avc," in *2009 IEEE International Symposium on Circuits and Systems*, May 2009, pp. 609–612.
- [8] Miguel Onofre Martínez-Rach, *Perceptual image coding for wavelet based encoders*, Ph.D. thesis, Universidad Miguel Hernández, Dec. 2014.
- [9] H. R. Sheikh and A. C. Bovik, "Image information and visual quality," *IEEE Transactions on Image Processing*, vol. 15, no. 2, pp. 430–444, Feb 2006.
- [10] Philippe Hanhart, "Vqmt: Video quality measurement tool," 3 2013.
- [11] G Bjontegaard, "Calculation of average psnr differences between rd-curves," *Proceedings of the ITU-T Video Coding Experts Group (VCEG) Thirteenth Meeting*, 01 2001.
- [12] K. N. Ngan, K. S. Leong, and H. Singh, "Adaptive cosine transform coding of images in perceptual domain," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 11, pp. 1743–1750, Nov 1989.
- [13] B. Chitprasert and K. R. Rao, "Human visual weighted progressive image transmission," *IEEE Transactions on Communications*, vol. 38, no. 7, pp. 1040–1044, July 1990.
- [14] N. Nill, "A visual model weighted cosine transform for image compression and quality assessment," *IEEE Transactions on Communications*, vol. 33, no. 6, pp. 551–557, June 1985.
- [15] J. Mannos and D. Sakrison, "The effects of a visual fidelity criterion of the encoding of images," *IEEE Transactions on Information Theory*, vol. 20, no. 4, pp. 525–536, July 1974.
- [16] Long-Wen Chang, Ching-Yang Wang, and Shiuh-Ming Lee, "Designing jpeg quantization tables based on human visual system," in *Proceedings 1999 International Conference on Image Processing (Cat. 99CH36348)*, Oct 1999, vol. 2, pp. 376–380 vol.2.
- [17] Yoshitaka Morigami Munsri Haque, Ali Tabatabai, "HVS model based default quantization matrices," in *Meeting report of the seventh meeting of the Joint Collaborative Team on Video Coding (JCT-VC)*, Nov 2011.
- [18] Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, April 2004.
- [19] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *The Thrity-Seventh Asilomar Conference on Signals, Systems Computers, 2003*, Nov 2003, vol. 2, pp. 1398–1402 Vol.2.