



UNIÓN INTERNACIONAL DE TELECOMUNICACIONES

UIT-T

SECTOR DE NORMALIZACIÓN
DE LAS TELECOMUNICACIONES
DE LA UIT

H.263

Apéndice III
(06/2001)

SERIE H: SISTEMAS AUDIOVISUALES Y
MULTIMEDIOS

Infraestructura de los servicios audiovisuales –
Codificación de imágenes vídeo en movimiento

Codificación de vídeo para comunicación a baja
velocidad binaria

**Apéndice III: Ejemplos de implementación del
codificador/decodificador H.263**

Recomendación UIT-T H.263 – Apéndice III

RECOMENDACIONES UIT-T DE LA SERIE H
SISTEMAS AUDIOVISUALES Y MULTIMEDIOS

CARACTERÍSTICAS DE LOS SISTEMAS VIDEOTELEFÓNICOS	H.100–H.199
INFRAESTRUCTURA DE LOS SERVICIOS AUDIOVISUALES	
Generalidades	H.200–H.219
Multiplexación y sincronización en transmisión	H.220–H.229
Aspectos de los sistemas	H.230–H.239
Procedimientos de comunicación	H.240–H.259
Codificación de imágenes vídeo en movimiento	H.260–H.279
Aspectos relacionados con los sistemas	H.280–H.299
SISTEMAS Y EQUIPOS TERMINALES PARA LOS SERVICIOS AUDIOVISUALES	H.300–H.399
SERVICIOS SUPLEMENTARIOS PARA MULTIMEDIOS	H.450–H.499

Para más información, véase la Lista de Recomendaciones del UIT-T.

Recomendación UIT-T H.263

Codificación de vídeo para comunicación a baja velocidad binaria

APÉNDICE III

Ejemplos de implementación del codificador/decodificador H.263

Resumen

El presente apéndice III a UIT-T H.263 tiene carácter informativo y contiene varios ejemplos de implementación del codificador y del decodificador, de interés para los usuarios de UIT-T H.263.

Orígenes

El apéndice III a la Recomendación UIT-T H.263, preparado por la Comisión de Estudio 16 (2001-2004) del UIT-T, fue aprobado por el procedimiento de la Resolución 1 de la AMNT el 8 de junio de 2001.

PREFACIO

La UIT (Unión Internacional de Telecomunicaciones) es el organismo especializado de las Naciones Unidas en el campo de las telecomunicaciones. El UIT-T (Sector de Normalización de las Telecomunicaciones de la UIT) es un órgano permanente de la UIT. Este órgano estudia los aspectos técnicos, de explotación y tarifarios y publica Recomendaciones sobre los mismos, con miras a la normalización de las telecomunicaciones en el plano mundial.

La Asamblea Mundial de Normalización de las Telecomunicaciones (AMNT), que se celebra cada cuatro años, establece los temas que han de estudiar las Comisiones de Estudio del UIT-T, que a su vez producen Recomendaciones sobre dichos temas.

La aprobación de Recomendaciones por los Miembros del UIT-T es el objeto del procedimiento establecido en la Resolución 1 de la AMNT.

En ciertos sectores de la tecnología de la información que corresponden a la esfera de competencia del UIT-T, se preparan las normas necesarias en colaboración con la ISO y la CEI.

NOTA

En esta Recomendación, la expresión "Administración" se utiliza para designar, en forma abreviada, tanto una administración de telecomunicaciones como una empresa de explotación reconocida de telecomunicaciones.

PROPIEDAD INTELECTUAL

La UIT señala a la atención la posibilidad de que la utilización o aplicación de la presente Recomendación suponga el empleo de un derecho de propiedad intelectual reivindicado. La UIT no adopta ninguna posición en cuanto a la demostración, validez o aplicabilidad de los derechos de propiedad intelectual reivindicados, ya sea por los miembros de la UIT o por terceros ajenos al proceso de elaboración de Recomendaciones.

En la fecha de aprobación de la presente Recomendación, la UIT ha recibido notificación de propiedad intelectual, protegida por patente, que puede ser necesaria para aplicar esta Recomendación. Sin embargo, debe señalarse a los usuarios que puede que esta información no se encuentre totalmente actualizada al respecto, por lo que se les insta encarecidamente a consultar la base de datos sobre patentes de la TSB.

© UIT 2002

Es propiedad. Ninguna parte de esta publicación puede reproducirse o utilizarse, de ninguna forma o por ningún medio, sea éste electrónico o mecánico, de fotocopia o de microfilm, sin previa autorización escrita por parte de la UIT.

ÍNDICE

Página

Apéndice III – Ejemplos de implementación del codificador/decodificador H.263.....	1
III.1 Introducción.....	1
III.2 Supuestos de aplicación.....	2
III.2.1 Codificación a velocidad binaria variable en canales de transmisión libres de error.....	2
III.2.2 Codificación a velocidad binaria fija para la transmisión por canales libres de errores (H.320 o H.324).....	3
III.2.3 Codificación a velocidad binaria fija para la transmisión por canales de conmutación de paquetes con pérdidas (H.323).....	3
III.2.4 Codificación a velocidad binaria fija para la transmisión por canales muy propensos a errores binarios (anexo C/H.324).....	6
III.3 Algoritmos comunes.....	7
III.3.1 Estimación del movimiento.....	7
III.3.2 Cuantificación.....	11
III.3.3 Modo VLC INTER alternativo (anexo S).....	13
III.3.4 Modo codificación INTRA avanzada (anexo I).....	13
III.3.5 Modo cuantificación modificada (anexo T).....	15
III.4 Algoritmos peculiares de determinados supuestos de aplicación.....	15
III.4.1 Decisión del modo.....	15
III.4.2 Control de velocidad.....	19
III.5 Posprocesamiento del decodificador.....	24
III.5.1 Posfiltrado de la sobreoscilación.....	25
III.5.2 Posfiltrado de desbloqueo.....	26
III.5.3 Detección de errores.....	28
III.5.4 Ocultación de errores.....	29
III.6 Sección de captura de información.....	29
III.6.1 El modo de tramas PB (anexos G y M).....	29
III.6.2 Modo escalabilidad (anexo O).....	30
III.6.3 Modo actualización de resolución reducida (anexo Q).....	33
III.6.4 Búsqueda rápida mediante desigualdades matemáticas.....	38
III.6.5 Control de velocidad en la trama de codificación.....	39
III.6.6 Observaciones sobre el uso optimizado del modo selección de imagen de referencia mejorada.....	42
III.7 Referencias.....	42

Recomendación UIT-T H.263

Codificación de vídeo para comunicación a baja velocidad binaria

APÉNDICE III

Ejemplos de implementación del codificador/decodificador H.263

III.1 Introducción

El presente apéndice describe el modelo de prueba a muy corto plazo (TMN, *test model near-term*) correspondiente a la versión 3 de UIT-T H.263, (incluidos sus anexos A a W) y asimismo el anexo X aprobado posteriormente. El modelo de prueba tiene por objeto explicar a los fabricantes la sintaxis y las especificaciones del decodificador de la Recomendación mediante ejemplos de implementación del codificador de vídeo.

La Recomendación UIT-T H.263 define una sintaxis de tren binario y su correspondiente decodificador de vídeo que permite el interfuncionamiento de terminales de vídeo de distintos fabricantes. El diseño del codificador de vídeo escapa al ámbito normativo de H.263, siendo responsabilidad del fabricante. No obstante, en la preparación de la Recomendación se determinó la conveniencia de favorecer ciertos métodos de codificación. Estos métodos proporcionaron buenos resultados en cuanto a calidad de vídeo y eficacia de la compresión y su grado de complejidad es adecuado para el funcionamiento con procesadores de uso general o especializados y con diferentes tipos de transmisión (por ejemplo en redes de conmutación de circuitos y en redes de conmutación de paquetes). Además, los niveles de calidad de funcionamiento obtenidos con estos métodos suelen servir de referencia para la investigación y la elaboración de futuras Recomendaciones del UIT-T sobre codificación de vídeo. El presente documento describe el funcionamiento del codificador y del decodificador así como las operaciones no contempladas en el texto normativo de H.263 aplicables, por ejemplo, al caso de violación de sintaxis provocada por la transmisión en canales propensos a errores.

Los documentos mencionados en este apéndice pertenecen a alguna de las siguientes categorías:

- Recomendaciones de la UIT: se consignan por sus conocidas abreviaturas, por ejemplo H.323 o BT.601. En ciertos casos la versión de la Recomendación es importante por lo que se consignará como proceda, indicando por ejemplo el año de publicación o el número de la versión. Las Recomendaciones del UIT-T pueden obtenerse directamente en el UIT-T. Para más información sírvase consultar a <http://www.itu.int>.
- Las contribuciones del Grupo de Expertos en codificación de vídeo de la Comisión de Estudio 16 del UIT-T: se consignan por sus abreviaturas, por ejemplo [Q15-D-58]. Para la mayor parte de las contribuciones citadas en el presente apéndice, se utilizan hiperenlaces hacia los documentos en un sitio ftp Q.6/16 oficioso que se encuentra actualmente en <ftp://standard.pictel.com/video-site>.
- Otras publicaciones académicas: se consignan mediante abreviaturas formadas principalmente por los nombres de los autores y las fechas de publicación, y pueden consultarse en las publicaciones citadas correspondientes.

La base del diseño H.263 es un codificador de vídeo de transformada híbrida con compensación de movimiento basado en bloques, compuesto de detector y compensador de movimiento, DCT, cuantificador, codificador de la longitud de la gama de repeticiones, codificador VLC y codificador FLC. Además hay varios modos de funcionamiento opcionales permitidos por la Recomendación H.263, definidos en sus anexos. El presente apéndice supone cierto conocimiento de la Recomendación H.263, sus modos opcionales, y la codificación de vídeo en general. En [GFS97]

hay una guía sobre la Recomendación H.263 versión 1 y sus modos opcionales; la publicación correspondiente a la versión 2 de la Recomendación H.263 se consigna en [Q15-D-58].

El presente apéndice describe métodos de codificación utilizados por unidades funcionales características de los codificadores. Este apéndice proporciona además información sobre la aplicación de diversos métodos a casos específicos, denominados supuestos de aplicación. A saber:

- codificación a velocidad binaria variable con valor de cuantificador fijo (véase 4.2.4/H.263) para la transmisión por canales libres de errores (como los que se suele utilizar en la investigación de la codificación de vídeo),
- codificación a velocidad binaria fija en un entorno prácticamente libre de errores (H.320/H.324),
- codificación a velocidad fija en un entorno de conmutación de paquetes con pérdidas (H.323), y
- codificación a velocidad binaria fija en un entorno muy propenso a errores binarios (anexo C/H.324).

La cláusula III.2 trata de los supuestos de aplicación y menciona los mecanismos descritos en las cláusulas subsiguientes. La cláusula III.2 define asimismo los entornos de simulación que se utilizan en este apéndice. Los mecanismos comunes a todos los supuestos de aplicación se describen en la cláusula III.3. Entre éstos se encuentra la búsqueda del vector de movimiento (MV, *motion vector*), la cuantificación y la utilización de los modos opcionales de H.263 que son comunes a todos los supuestos de aplicación. Los mecanismos propios de supuestos de aplicación específicos se tratan en la cláusula III.4. Junto con la propia Recomendación, estas cláusulas definen un marco para un códec H.263 que exhibe un comportamiento bastante aceptable en la mayoría de los supuestos de aplicación. La cláusula III.5 profundiza en el posprocesamiento del decodificador. La cláusula III.6 reúne toda la información que ha sido considerada de interés para su inclusión en el presente apéndice aunque no encaje exactamente en ninguno de los supuestos de aplicación.

III.2 Supuestos de aplicación

Esta cláusula trata de los supuestos de aplicación de la codificación de vídeo en base a H.263. Pretende ofrecer una visión general de los mecanismos útiles para cada supuesto y bosquejar, cuando proceda, el correspondiente supuesto de simulación. La descripción del entorno exterior al trayecto de datos de vídeo se ha reducido al mínimo aunque se han consignado algunas referencias oportunas.

III.2.1 Codificación a velocidad binaria variable en canales de transmisión libres de error

El supuesto de codificación a velocidad binaria variable utiliza un valor cuantificador constante para todas las imágenes y sus regiones a fin de obtener un nivel de calidad constante. Este supuesto es apropiado para la investigación de la codificación de vídeo y los trabajos de normalización asociados. Por ejemplo, proporciona un marco para evaluar la calidad objetiva y subjetiva de los trenes binarios generados de acuerdo con las nuevas soluciones para mejorar el rendimiento de la codificación. Este supuesto corresponde a dos modelos distintos: el modelo de baja complejidad y el de alta calidad. Ambos modelos se describen en las dos subcláusulas siguientes.

III.2.1.1 Modelo de codificación de baja complejidad

El modelo de codificación de baja complejidad utiliza los modos opcionales definidos en el perfil 1 del anexo X de H.263, a saber: el modo codificación INTRA avanzado (anexo I), el modo filtro de desbloqueo (anexo J), el modo cuantificación modificada (anexo T) y una parte del modo información de mejora suplementaria (anexo L). Hay que señalar que este apéndice no contempla las capacidades mejoradas del modo información de mejora suplementaria. No obstante, sí se contemplan ciertas características que podrían ser importantes para el diseño de ciertos productos. La estimación del movimiento y la decisión del modo se efectúan por medio de la búsqueda del vector de movimiento de baja complejidad y los métodos de decisión del modo. Hay más información al

respecto en las cláusulas III.3.1.2 (búsqueda del vector de movimiento de baja complejidad), III.4.1.2 (decisión de modo de baja complejidad), III.3.4 (modo codificación INTRA avanzado) y III.3.5 (modo cuantificación modificada).

III.2.1.2 Modelo de codificación de alta calidad

El modelo de codificación de alta calidad se ha diseñado para mejorar la calidad de la imagen reconstruida respecto al modelo de baja complejidad incurriendo en una mayor complejidad de cálculo. El modelo de codificación de alta calidad utiliza los modos opcionales definidos en el perfil 5 del anexo X de H.263, entre los que se encuentran los modos del perfil 1 utilizados por el modelo de codificación de baja complejidad así como el modo vector de movimiento sin restricción (anexo D), el modo predicción avanzada (anexo F), y el modo selección de imagen de referencia mejorada (anexo U) cuando se utilizan varias imágenes de referencia en la compensación del movimiento. La estimación del movimiento y la decisión del modo se realizan por medio del método de búsqueda del vector de movimiento con codificación de alta calidad y del método de decisión del modo. Para más información pueden consultarse las cláusulas III.3.1.4 (búsqueda del vector de movimiento con codificación de alta calidad), III.4.1.3 (decisión del modo con codificación de alta calidad), III.3.4 (modo codificación INTRA avanzada), III.3.5 (modo cuantificación modificada) y III.4.2.4 (decisión del modo combinado, búsqueda del vector de movimiento y selección de la imagen de referencia mejorada).

III.2.2 Codificación a velocidad binaria fija para la transmisión por canales libres de errores (H.320 o H.324)

Esta aplicación se caracteriza por la necesidad de conseguir un objetivo de velocidad binaria fija con un retardo razonablemente bajo. El mecanismo de transporte está orientado a bits y proporciona un entorno que puede considerarse libre de errores en todos los casos prácticos. Este supuesto está más cerca de una aplicación de carácter práctico que el anterior, y por lo tanto el compromiso entre complejidad y calidad adquiere gran importancia. Para alcanzar el objetivo de velocidad binaria, el tamaño del paso cuantificador deja de ser fijo aunque sigue estando definido por el algoritmo de control de velocidad. Además, aunque suele especificarse un objetivo de velocidad de trama, el algoritmo de control de velocidad puede omitir imágenes fuente libremente, cuando se sobrepasa la velocidad binaria prevista.

III.2.2.1 Modelo de codificación de baja complejidad

El modelo de baja complejidad consta de todo el mecanismo descrito en III.2.1.1, más un control de velocidad a fin de alcanzar el objetivo de velocidad binaria. Puede obtenerse más información en III.3.1.2 (búsqueda del vector de movimiento de baja complejidad), III.4.1.2 (decisión del modo de baja complejidad), III.3.4 (modo codificación INTRA avanzada), III.3.5 (modo cuantificación modificada) y III.4.2 (control de velocidad).

III.2.2.2 Modelo de codificación de alta calidad

El modelo de alta calidad consta de los mecanismos descritos en III.2.1.2, más un control de velocidad para alcanzar el objetivo de velocidad binaria. Cuando se combina el control de velocidad con la estimación del movimiento de alta calidad y los algoritmos de decisión de modo este apéndice introduce ciertas simplificaciones que se describen en III.4.2.3.

III.2.3 Codificación a velocidad binaria fija para la transmisión por canales de conmutación de paquetes con pérdidas (H.323)

Este supuesto de aplicación se caracteriza por la necesidad de alcanzar un objetivo de velocidad binaria y un método de empaquetado para el transporte. En los sistemas H.323, se utiliza un sistema de transporte orientado a paquetes basado en RTP [RFC 1889]. Las principales características de este transporte son las siguientes:

- tamaño variable de los paquetes, definido por el remitente, de 1,5 kbyte, aproximadamente, para reducir la tara de empaquetado y ajustarse al tamaño de la unidad de transferencia máxima (MTU, *maximum transfer unit*) de Internet, e
- índice elevado de pérdida de paquetes.

Obsérvese que el codificador tiene acceso a los promedios a medio plazo de los índices de pérdidas de paquetes, a partir de los informes obligatorios del receptor RTCP que forman parte del RTP. Como el RTP y las capas de protocolos subyacentes asociadas garantizan la entrega de paquetes libres de errores en la secuencia correcta¹, la única fuente de errores la constituye la pérdida de paquetes.

Ya se sabe que la comunicación bidireccional con mensajes por el canal de respaldo puede mejorar considerablemente la calidad de la imagen reproducida. No obstante, la comunicación bidireccional suele ser inviable debido al posible carácter multidifusión del transporte y de la aplicación y asimismo a las posibles restricciones de la aplicación sobre el retardo de la transmisión. Por otra parte, la complejidad de la simulación de un entorno realmente bidireccional no es despreciable. Por todas estas razones, en adelante sólo se considerará la comunicación unidireccional.

En primer lugar, se utilizará un algoritmo de control de velocidad para velocidades binarias fijas, con objeto de simplificar la interacción entre el control de la velocidad origen y los mecanismos de transporte. En los sistemas reales, aunque la gestión de la memoria intermedia del RTP puede suavizar las variaciones a corto plazo de las velocidades de transmisión, el objetivo de velocidad binaria y el tamaño de la memoria intermedia del receptor se suelen ajustar periódicamente con arreglo a factores tales como el índice medio de pérdida de paquetes y los límites impuestos por la facturación. En segundo lugar, se supone la existencia de índices medios constantes de pérdidas de paquetes, a fin de simplificar la interacción entre la capacidad de recuperación de errores y los mecanismos de transporte. En los sistemas reales el soporte de la capacidad de recuperación de errores debe ser adaptable, por ejemplo en base a índices medios de pérdida de paquetes en un intervalo de tiempo.

El resto de esta cláusula trata de cuestiones relativas al empaquetado y desempaquetado y de la aplicación de instrumentos de codificación de vídeo a este supuesto. Existen dos modelos distintos, uno de baja complejidad y otro de alta calidad.

III.2.3.1 Empaquetado y desempaquetado

Esta subcláusula describe un sistema de empaquetado y desempaquetado, que utiliza RFC 2429 [RFC 2429] como formato de la parte útil del RTP, de funcionamiento comprobado en el entorno propio de este supuesto.

Se supone la existencia de un mecanismo de codificación para reducir la propagación de errores temporales, inevitables en un entorno de pérdida de paquetes. Un mecanismo similar, descrito en III.4.1.1, consiste en la utilización razonable de macrobloques codificados INTRA. Se supone asimismo la utilización del modo estructura en rebanada (anexo K), con rebanadas del mismo tamaño y forma que el grupo de bloques (GOB). En el presente supuesto se utilizan rebanadas en vez de GOB porque el anexo K permite una estructura de tren binario tal que las rebanadas no tienen por qué aparecer en el orden de exploración normal. El plan de empaquetado depende de dicha configuración.

El plan de empaquetado se basa en la intercalación de rebanadas con forma de GOB de numeración par e impar, y nace de dos consideraciones de diseño. En primer lugar, como la tara de empaquetado para las cabeceras IP/UDP/RTP de Internet es de 40 bytes por paquete aproximadamente, hay que

¹ Aunque el RTP no garantiza la correcta numeración de la secuencia como funcionalidad del protocolo, su cabecera contiene un número de secuencia que puede utilizarse para verificar la correcta secuenciación de los paquetes.

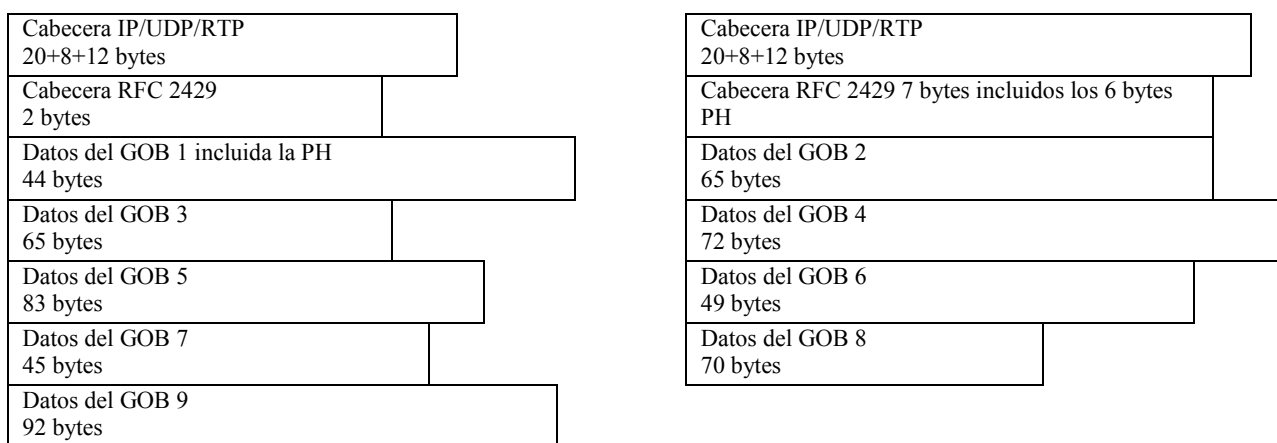
utilizar paquetes bastante grandes. En segundo lugar, para poder compensar eficazmente los errores, como explica III.5.4, no deben colocarse rebanadas consecutivas dentro del mismo paquete. En consecuencia hay dos paquetes por imagen lo que permite compensar razonablemente la pérdida de macrobloques siempre que se pierda solamente uno de los dos paquetes. Este método puede ampliarse a más de dos paquetes por imagen siempre que el tamaño de la imagen codificada sea superior a 2800 bytes, suponiendo un tamaño máximo de tara de 1400 bytes por paquete y un tamaño de MTU de 1500 bytes.

La pérdida de contenido de la cabecera de imagen puede degradar gravemente la calidad de la imagen decodificada. Esto puede ocultarse suponiendo que el contenido de la cabecera de imagen permanece inalterado, excepto en el campo de referencia temporal, que puede recuperarse de la referencia temporal que contiene la cabecera RTP. Cuando el codificador de vídeo modifica el contenido de la cabecera de imagen, se utiliza un mecanismo RFC 2429 que permite introducir un duplicado de la cabecera de imagen en la cabecera de la parte útil de cada paquete, para poder efectuar una decodificación parcial de una imagen (y la subsiguiente ocultación) cuando se recibe un único paquete.

La figura III.1 muestra un ejemplo característico del plan de empaquetado para un tren binario de vídeo codificado a 50 kbit/s y 10 tramas por segundo con resolución QCIF. La tara de empaquetado constante (compuesta por las cabeceras IP/UDP/RTP, que supone un total de 40 bytes por paquete) es de 80 bytes por imagen empaquetada. La palabra GOB de la figura se refiere a la rebanada con forma de GOB (grupo de bloques) cuya posición espacial corresponde al GOB del número asociado.

Paquete 1: contiene la cabecera de la imagen original y los GOB impares.
Tamaño total: 371 bytes

Paquete 2: contiene un duplicado de la cabecera de imagen y los GOB pares.
Tamaño total: 303 bytes



PH Cabecera de imagen H.263

Figura III.1/H.263 – Ejemplo de los dos paquetes de una imagen con el sistema de empaquetado intercalado

La cabecera mínima de 2 bytes definida en RFC 2429 no aumenta la tara porque sustituye al código de inicio de imagen de 16 bits o código de inicio de rebanada que precede a cada imagen/rebanada H.263.

Si se utiliza este sistema de empaquetado en un entorno de pérdida de paquetes, pueden presentarse cuatro situaciones distintas durante el desempaqueado y la reconstrucción. Cuando se reciben ambos paquetes la decodificación es inmediata. Cuando sólo se recibe el primer paquete, se decodifican los datos disponibles y las rebanadas perdidas se detectan y se ocultan, por ejemplo como se explica en III.5.4. Cuando sólo se dispone del segundo paquete, el duplicado de la cabecera de imagen y los datos útiles se concatenan y se decodifican. Se detectan las rebanadas perdidas, ya que provocan

violaciones de sintaxis, y se compensan de acuerdo con lo descrito en III.5.4. Si se pierden ambos paquetes, el decodificador no dispone de datos y se oculta toda la imagen volviendo a mostrar la imagen anterior.

III.2.3.2 Modelo de codificación de baja complejidad

El modelo de codificación de baja complejidad utiliza los mismos mecanismos de codificación que los descritos en III.2.2.1 con dos adiciones que mejoran la capacidad de recuperación de errores. Se utilizan rebanadas con forma de GOB, y se insertan cabeceras al principio de cada rebanada que funcionan como marcas de sincronización y permiten restaurar la codificación predictiva espacial, tal como la codificación del vector de movimiento y la codificación intercoeficientes definidas en el anexo K de H.263. Por otra parte, la velocidad a la que se fuerza la codificación de los macrobloques en el modo INTRA (actualizado) varía con la inversa del índice medio de pérdida de paquetes a medio plazo. Esto se implementa mediante el Frecuencia_Renovación_MB_INTRA del mecanismo de actualización de macrobloques INTRA descrito en III.4.1.1.

III.2.3.3 Modelo de codificación de alta calidad

El modelo de codificación alto utiliza el empaquetado, la ocultación de errores y las rebanadas con forma de GOB descritas anteriormente. Se utiliza asimismo una ampliación del modo alta calidad descrito en III.2.1.2 que permite sustituir la decisión del modo de codificación de alta calidad de III.4.1.3 por la de III.4.1.5. Esta decisión del modo está motivada por la propensión a errores de la red, el proceso de empaquetado, la velocidad de pérdida de paquetes y la ocultación de errores utilizada por el decodificador para seleccionar el modo de codificación de macrobloques. Puede consultarse información detallada en III.4.1.5 (decisión del modo de alta calidad para entornos propensos a errores).

III.2.4 Codificación a velocidad binaria fija para la transmisión por canales muy propensos a errores binarios (anexo C/H.324)

Este supuesto de aplicación se caracteriza por la necesidad de conseguir un objetivo de velocidad binaria fija en un entorno muy propenso a errores binarios. En este entorno, el protocolo de transporte H.223 empleado por el correspondiente sistema anexo C/H.324 se optimiza principalmente para el funcionamiento con bajo retardo. H.223 no puede asegurar la entrega libre de errores binarios de la parte útil, aunque utilice los algoritmos de retransmisión opcionales. Por consiguiente, el decodificador de vídeo debe poder detectar y manipular los errores en los bits.

Por razones prácticas, se necesita adoptar ciertas hipótesis e introducir algunas simplificaciones relativas a la simulación del transporte:

- Se utiliza el modo de tramas de H.223, con SDU AL3 para los datos H.263 (teniendo en cuenta la detección de errores mediante CRC de 16 bits).
- Se supone que la comunicación es unidireccional, o sea, que no se utilizan algoritmos de retransmisión ni mecanismos de canal de respaldo, debido a los estrictos límites del retardo.

Se definen dos modelos: uno de codificación de baja complejidad que utiliza exclusivamente las herramientas de codificación disponibles en la versión 2 (1998) de H.263 y un modelo de codificación de alta capacidad de recuperación de errores, que contiene las herramientas de la versión 3 de H.263.

III.2.4.1 Modelo de codificación de baja complejidad

El modelo de baja complejidad utiliza las mismas herramientas de codificación de vídeo que el modelo de baja complejidad de velocidad binaria fija en entorno libre de errores, descrito en III.2.2.1. Para mejorar la capacidad de recuperación de errores, se incluye asimismo un método de empaquetado y otro de codificación INTRA forzada. Cada uno de los grupos de bloques (GOB) se codifica con una cabecera que sirve de marca de sincronización y restaura la codificación predictiva espacial, tal como la codificación del vector de movimiento y la codificación intercoeficientes,

definidos en H.263. Cada uno de los GOB codificados se empaqueta en una SDU AL3. Las SDU AL3 recibidas que no superen la prueba CRC no son procesadas por el decodificador, sino que se ocultan. Se evita la decodificación parcial de los GOB corrompidos por la dificultad de definir con exactitud la operación del decodificador en dicho caso. Además, el problema se complica por el hecho de que los errores no detectados pueden provocar perturbaciones muy visibles tales como bloques de distinto color. La única violación de sintaxis permitida, que se utiliza para detectar GOB perdidos en el decodificador, es la de pérdida de secuencia de la numeración de los GOB.

Se utiliza una frecuencia fija de renovación de los macrobloques INTRA. Para determinar la frecuencia de renovación de los macrobloques INTRA, se utiliza la siguiente regla práctica. Cada $1/p$ -avo (redondeado al entero más próximo) de tiempo que un cierto macrobloque se codifica con coeficientes, ha de codificarse en modo INTRA, siendo p la probabilidad media de pérdida para todos los macrobloques de una secuencia correspondientes a una determinada característica de error. Es decir, si la probabilidad de pérdida calculada es 0,1, cada $1/10$ de tiempo hay que codificar un macrobloque con información de coeficientes en modo INTRA. Este algoritmo debe implementarse mediante el Frecuencia_Renovación_MB_INTRA del mecanismo de actualización de macrobloques INTRA, descritos en III.4.1.1. A diferencia del algoritmo recomendado en 5.1.4.3 de la versión 2 de H.263, el bit Tipo de Redondeo PLUSPTYPE se pone a "0" independientemente del tipo de imagen [Q15-I-26].

III.2.4.2 Modelo de codificación de alta capacidad de recuperación de errores

El modelo de codificación de alta capacidad de recuperación de errores se basa en el modo rebanada compartición de datos, del que trata la sección III.4.2.4. Además, se utiliza el mecanismo "repetición de la anterior cabecera de imagen" del anexo W, del que trata la sección III.4.2.6.

III.3 Algoritmos comunes

Esta cláusula trata de los algoritmos comunes a todos los supuestos de aplicación.

III.3.1 Estimación del movimiento

En el presente apéndice se describen tres algoritmos de estimación de movimiento:

- Un algoritmo de búsqueda rápida de baja complejidad para un número reducido de posiciones de búsqueda.
- Un algoritmo de búsqueda completa de complejidad media.
- Un algoritmo de búsqueda completa de alta complejidad que tiene en cuenta la velocidad binaria resultante del vector de movimiento además de la calidad de la concordancia.

De estos tres algoritmos los más utilizados son el de baja calidad y el de alta calidad, en los correspondientes modelos de alta y baja calidad de los supuestos de aplicación. El algoritmo de calidad media puede sustituir al algoritmo de baja complejidad, ofreciendo una calidad velocidad-distorsión ligeramente menor con una complejidad de cálculo muy superior. Obsérvese que puede utilizarse cualquier algoritmo de búsqueda basado en bloques. No obstante los tres algoritmos descritos en el modelo de prueba se han utilizado muchas veces y se ha comprobado su funcionamiento en los distintos supuestos de aplicación.

H.263 puede utilizar uno o cuatro vectores de movimiento por macrobloque en función de los modos opcionales activos. Pueden consultarse los anexos D, F y J de H.263 donde se describen las diversas limitaciones de los vectores de movimiento. Por otra parte, la amplitud admisible del intervalo de búsqueda depende del submodo del anexo D que se utilice.

III.3.1.1 Medida de la distorsión de la suma de valores absolutos de las diferencias

Los algoritmos de búsqueda del vector de movimiento de píxel entero y de medio píxel emplean la suma de valores absolutos de las diferencias (SAD, *sum of absolute difference*) como medida de la

distorsión. La SAD se calcula sobre todos los píxeles de luminancia de los macrobloques candidatos y objetivos. En ciertos casos el algoritmo de decisión del modo se inclinará por la SAD del vector (0,0). La SAD de un vector de movimiento candidato se calcula de la siguiente manera:

$$SAD(u, v) = \sum_{j=0}^{M-1} \sum_{i=0}^{M-1} |\tilde{X}_{t-1}(i+u, j+v)|$$

siendo:

$$M = \begin{cases} 8 & \text{para vectores de movimiento } 8 \times 8 \\ 16 & \text{para vectores de movimiento } 16 \times 16 \end{cases}$$

- X_t la trama objetivo
- \tilde{X}_{t-1} la anterior trama reconstruida
- (i, j) la posición espacial dentro de la trama objetivo
- (u, v) el vector de movimiento candidato

Para reducir la complejidad del cálculo de la SAD se acude a la técnica de distorsión parcial. La técnica de distorsión parcial compara la SAD acumulada tras cada fila de M píxeles con la SAD mínima hasta ese momento dentro del intervalo de búsqueda. Si la SAD acumulada es mayor que la SAD mínima hasta ese momento, se da por terminado el cálculo del vector movimiento candidato, ya que éste no encontrará una coincidencia mejor, es decir un valor de la SAD menor, que la mejor coincidencia hasta el momento.

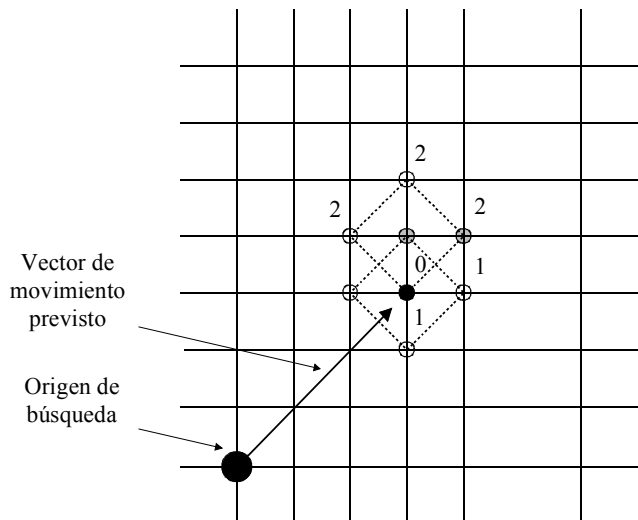
III.3.1.2 Búsqueda de baja complejidad

La búsqueda de baja complejidad se aprobó una vez realizadas comparaciones exhaustivas con el algoritmo de búsqueda completa que pusieron de manifiesto la introducción de escasa o nula degradación de la calidad de funcionamiento para las imágenes de resolución QCIF y CIF. Puede consultarse información detallada en [GCK99] y los resultados exhaustivos de la comparación de la calidad de funcionamiento con el algoritmo de búsqueda completa en [Q15-B-23].

III.3.1.2.1 Procedimiento de búsqueda para bloques de 16×16 y precisión de píxel entero

El centro de búsqueda es el vector mediano de movimiento previsto, definido en 6.1.1 y F.2 de H.263. Se buscará asimismo el vector (0,0) cuando sea distinto del vector de movimiento previsto. Se favorece el vector (0,0) restando 100 de la SAD calculada. El intervalo de búsqueda admisible viene definido por el submodo del anexo D utilizado.

El algoritmo busca secuencialmente capas rómbicas cada una de las cuales contiene los cuatro vecinos más próximos al centro de la capa actual. A continuación la capa $i + 1$ se centra en el punto de SAD mínima de la capa i . De este modo las capas sucesivas tienen distintos centros y contienen como máximo tres vectores de movimiento candidatos sin probar, salvo la primera capa en torno al vector de movimiento previsto que contiene cuatro vectores de movimiento candidatos sin probar. La figura III.2 muestra un ejemplo de esta búsqueda.



T1609820-01

- 0 Posición de búsqueda de la capa 0 (vector de movimiento previsto)
- 1 Posición de búsqueda de la capa 1
- 2 Posición de búsqueda de la capa 2

NOTA – Los puntos sombreados representan las SAD mínimas de las correspondientes capas.

Figura III.2/H.263 – Ejemplo de búsqueda rápida de píxel entero cubriendo dos capas desde el vector de movimiento previsto

La búsqueda sólo se detiene una vez que:

- 1) ya se han examinado todos los vectores de movimiento candidatos de la capa actual y el valor de la SAD mínima de la capa actual es superior al de la capa anterior; o
- 2) la búsqueda ha llegado al límite de la zona de búsqueda admisible y se intenta superar dicho límite.

Posteriormente el vector de movimiento de píxel entero de 16×16 se afina a medio píxel como se describe a continuación.

III.3.1.2.2 Afinación a medio píxel de 16×16

El área de búsqueda de medio píxel es de ± 1 medio píxel en torno al mejor candidato de 16×16 . La búsqueda se realiza calculando valores de medio píxel como se explica en 6.1.2/H.263 y calculando a continuación la SAD correspondiente a cada uno de los vectores de medio píxel posibles. De este modo se comprueban ocho vectores candidatos adicionales. Se tomará el vector resultante de la mejor concordancia durante la afinación de medio píxel.

III.3.1.2.3 Búsqueda de píxel entero de 8×8

Los vectores de movimiento entero de 8×8 se igualan al mejor vector de movimiento de píxel entero de 16×16 , no habiendo búsqueda adicional de píxel entero de 8×8 . Debido a la codificación diferencial de los vectores de movimiento, esta restricción garantiza que los vectores de movimiento de 8×8 seleccionados para el macrobloque objetivo están fuertemente correlacionados.

III.3.1.2.4 Afinación a medio píxel de 8×8

La afinación a medio píxel se aplica a cada uno de los bloques que rodean al vector de píxel entero de 8×8 . La búsqueda es esencialmente idéntica a la descrita en III.3.1.2.2, pero con tamaño de bloque igual a 8. Obsérvese que los vectores (0,0) no quedan favorecidos en este caso.

III.3.1.3 Búsqueda completa de calidad media

III.3.1.3.1 Búsqueda de píxel entero de 16×16

El centro de búsqueda vuelve a ser el vector de movimiento mediano previsto, definido en 6.1.1 y F.2 de H.263. Se busca también el vector (0,0) cuando sea distinto del vector de movimiento previsto. Se favorece el vector (0,0) restando 100 de la SAD calculada. El intervalo de búsqueda admisible viene impuesto por el submodo del anexo D que se utilice.

El algoritmo efectúa una búsqueda secuencial de las capas de acuerdo con un patrón espiral centrífugo con respecto al vector de movimiento previsto hasta alcanzar el extremo del intervalo de búsqueda admisible. Esto ocurre tanto en la búsqueda de píxel entero de 16×16 como en la de 8×8 . De este modo cada capa, salvo la que contiene exclusivamente el origen de la búsqueda, contribuye con $8 \times \text{número_de_capa}$ vectores de movimiento candidatos. En cada capa, la búsqueda comienza en el extremo superior izquierdo de la espiral y continúa en el sentido de las agujas del reloj.

La búsqueda en espiral es más eficaz cuando comienza en el vector de movimiento mediano previsto porque normalmente se producirá una buena coincidencia en poco tiempo, y si se combina con la técnica de distorsión parcial se puede reducir significativamente el número SAD de filas calculadas. La afinación a medio píxel es idéntica a la del algoritmo de baja complejidad descrito en III.3.1.2.2.

III.3.1.3.2 Búsqueda de píxel entero de 8×8

La búsqueda de píxel entero de 8×8 se realiza de modo exhaustivo en una ventana de búsqueda reducida, en el entorno del mejor vector de movimiento de píxel entero de 16×16 , con un tamaño de ± 2 unidades de píxel entero. Esto permite una variación ligeramente mayor en los vectores de movimiento de 8×8 del macrobloque objetivo, aunque estos vectores pueden presentar en este caso concordancias de superior calidad. La afinación a medio píxel es idéntica a la del algoritmo de baja complejidad descrito en III.3.1.2.4.

III.3.1.4 Búsqueda completa de alta calidad con optimización de la distorsión de velocidad

Para regularizar el problema de la estimación del movimiento se utiliza una formulación Lagrangiana en la que la distorsión se pondera con la velocidad por medio de un multiplicador de Lagrange. Cuando se activen modos que soporten los vectores de movimiento de 8×8 , por ejemplo los del anexo F y/o los del anexo J se realizará una búsqueda completa de los vectores de movimiento de píxel entero de 16×16 y de 8×8 . En caso contrario, sólo se realizará la búsqueda completa de los vectores de movimiento de píxel entero de 16×16 .

En cada uno de los macrobloques de 16×16 o de los bloques de 8×8 , la búsqueda del vector de movimiento de píxel entero emplea el patrón de búsqueda espiral descrito en III.3.1.3.1. La afinación a medio píxel utiliza el patrón descrito en III.3.1.2.2. El vector de movimiento de medio píxel correspondiente al bloque i de 8×8 se obtiene antes de continuar calculando los vectores de movimiento de píxel medio o entero para el bloque $i + 1$ de 8×8 , etc. Esto permite calcular con exactitud los términos de velocidad correspondientes a los vectores de movimiento de píxel entero y medio de los bloques 8×8 , ya que el vector de movimiento previsto estará completamente determinado.

La búsqueda de píxel entero con optimización de la distorsión de velocidad selecciona el vector de movimiento que minimiza el coste de Lagrange definido como:

$$J = D + \lambda_{\text{movimiento}} R$$

La distorsión, D , se define como la SAD entre el componente de luminancia del macrobloque (o bloque) objetivo y el macrobloque (o bloque) de la imagen de referencia desplazado por el vector de movimiento candidato. Los cálculos de la SAD utilizan los criterios de concordancia de distorsión parcial con la SAD mínima encontrada hasta el momento. La velocidad R , se define como la suma de la velocidad correspondiente a los vectores de movimiento candidatos del macrobloque (o bloque)

verticales y horizontales, teniendo en cuenta el vector de movimiento previsto definido en 6.1.1/H.263. La selección del parámetro λ se explica más adelante. La búsqueda se centra en el vector de movimiento previsto. El vector (0,0) se busca aunque no se favorece dado que la minimización Lagrangiana ya tiene en cuenta la velocidad del vector de movimiento. A continuación se afinan a medio píxel los mejores candidatos de 16×16 y 8×8 , por medio de una búsqueda de ± 1 medio píxel. La búsqueda de medio píxel utiliza la formulación Lagrangiana anterior para seleccionar el vector de movimiento de medio píxel que minimice el coste de Lagrange. A continuación el algoritmo de decisión del modo utiliza los mejores vectores de 16×16 y 8×8 .

La elección de $\lambda_{movimiento}$ tiene muy poca repercusión en el resultado de la estimación del movimiento del bloque de 16×16 . Sin embargo, el resultado correspondiente a los bloques de 8×8 depende mucho de $\lambda_{movimiento}$, que se calcula como

$$\lambda_{movimiento} = 0,92 \cdot QP$$

siendo QP el parámetro de cuantificación del macrobloque [Q15-D-13]. Cuando la aplicación es a velocidad binaria fija, la relación queda modificada como se describe en III.4.2.3.

III.3.1.5 Otras cuestiones relativas a la búsqueda del vector de movimiento

Aunque no se utiliza en ningún supuesto de aplicación, el modelo de prueba contiene algoritmos de búsqueda del vector de movimiento para tramas PB (cuando rigen los anexos G o M), en el modo actualización de resolución reducida, y para los diversos tipos de imagen admisibles con el modo escalabilidad definido en el anexo O. Éstos se tratan con más detalle en III.6.1.1, III.6.3 y III.6.2.2 respectivamente.

III.3.2 Cuantificación

El parámetro de cuantificación $QUANT$ puede tomar valores enteros entre 1 y 31. El espaciado de reconstrucción de la cuantificación para coeficientes no nulos es $2 \cdot QP$, siendo:

$QP = 4$ para los coeficientes DC INTRA cuando no se está en modo codificación INTRA avanzada, y

$QP = QUANT$ en los demás casos.

Definiciones:

COF coeficiente de la transformada (o coeficiente diferencia) a cuantificar

$LEVEL$ versión cuantificada del coeficiente de la transformada

REC valor del coeficiente reconstruido

"/" división con truncamiento

La regla básica de reconstrucción inversa de la cuantificación para todos los coeficientes no nulos cuantificados puede expresarse como:

$$|REC| = QP \cdot (2 \cdot |LEVEL| + p) \quad \text{si } QP = \text{"impar"}; \text{ y}$$

$$|REC| = QP \cdot (2 \cdot |LEVEL| + p) - p \quad \text{si } QP = \text{"par"},$$

siendo:

$p = 1$ para los coeficientes INTER,

$p = 1$ para los coeficientes INTRA no DC cuando no se está en modo codificación INTRA avanzada,

$p = 0$ para los coeficientes INTRA DC cuando no se está en modo codificación INTRA avanzada, y

$p = 0$ para los coeficientes INTRA (tanto DC como no DC) cuando se está en el modo codificación INTRA avanzada.

El parámetro p es la unidad cuando el espaciado del valor de la reconstrucción no es uniforme (o sea, cuando hay una expansión del espacio de reconstrucción en torno a 0) y p es 0 en los demás casos. La regla de cuantificación del codificador aplicable se compensa con el efecto de p sobre el espacio de reconstrucción. Para que la cuantificación presente un error cuadrático medio óptimo, la separación entre los umbrales de decisión de la cuantificación debe ser tal que los valores de reconstrucción formen un centroide de valores esperados en cada región. Si la función de densidad de probabilidad (pdf, *probability density function*) de los coeficientes se aproxima por la distribución de Laplace, el mismo desplazamiento sencillo para todos los intervalos de cuantificación puede lograr esta separación óptima. Los coeficientes se cuantifican de acuerdo con dicha regla, es decir utilizan una forma entera de:

$$|LEVEL| = \left[|COF| + (f - p) \cdot QP \right] / (2 \cdot QP)$$

siendo:

$f \in \left\{ \frac{1}{2}, \frac{3}{4}, 1 \right\}$ un parámetro que se utiliza para situar los umbrales de decisión del cuantificador de modo que todos los valores de reconstrucción queden comprendidos entre el resultado de redondear por exceso al entero más próximo ($f = 1$) y el de la reconstrucción de borde izquierdo ($f = 0$) donde f se escoge de acuerdo con la pendiente media (exponencial) de bajada de la pdf del origen en cada paso no nulo.

III.3.2.1 Cuantificación de los coeficientes INTER

Los coeficientes INTER (sean DC o no) se cuantifican de acuerdo con:

$$|LEVEL| = \left(|COF| - QUANT / 2 \right) / (2 \cdot QUANT)$$

lo que corresponde a $f = \frac{1}{2}$ y $p = 1$.

III.3.2.2 Cuantificación para los coeficientes INTRA no DC cuando no se está en modo codificación INTRA avanzada

Los coeficientes INTRA no DC cuando no se está en el modo codificación INTRA avanzada se cuantifican del siguiente modo:

$$|LEVEL| = |COF| / (2 \cdot QUANT)$$

lo que corresponde a $f = 1$ y $p = 1$.

III.3.2.3 Cuantificación para coeficientes INTRA DC cuando no se está en el modo de codificación INTRA avanzada

El coeficiente DC de un bloque INTRA cuando no se está en el modo codificación INTRA avanzada se cuantifica de acuerdo con:

$$LEVEL = (COF + 4) / (2 \cdot 4)$$

Esto corresponde a $f = 1$ y $p = 0$. Obsérvese que, en este caso, COF y $LEVEL$ nunca son negativas y que QP es siempre 4.

III.3.2.4 Cuantificación para coeficientes INTRA cuando se está en el modo codificación INTRA avanzada

Los coeficientes INTRA cuando se está en el modo codificación INTRA avanzada (ya sea DC o no) se cuantifican de acuerdo con:

$$|LEVEL| = (|COF| + 3 \cdot QUANT / 4) / (2 \cdot QUANT)$$

Esto corresponde a $f = \frac{3}{4}$ y $p = 0$.

III.3.3 Modo VLC INTER alternativo (anexo S)

Durante la codificación de entropía, el codificador utilizará el cuadro VLC INTRA del anexo I para codificar los bloques INTER siempre que se cumplan los tres criterios siguientes:

- Se utiliza el modo VLC INTER alternativo del anexo S y esto se indica en la cabecera de imagen.
- La codificación de los coeficientes de los bloques INTER mediante el cuadro VLC INTRA del anexo I genera menos bits que la codificación mediante el cuadro VLC INTER.
- La utilización del cuadro VLC INTRA del anexo I deberá ser detectable por el decodificador. El decodificador supondrá que los coeficientes se han codificado con el cuadro VLC INTER. El codificador detectará la utilización del cuadro VLC INTRA del anexo I cuando se direccionen coeficientes fuera del intervalo de 64 coeficientes de los bloques 8×8 .

Esto ocurrirá fácilmente cuando haya muchos coeficientes grandes, gracias al diseño del VLC INTRA, y supone un ahorro importante en número de bits a velocidades binarias altas.

III.3.4 Modo codificación INTRA avanzada (anexo I)

La codificación INTRA avanzada es un método de mejora de la codificación de bloques INTRA que utiliza la predicción de bloques INTER. La aplicación de esta técnica se describe en el anexo I de H.263. El procedimiento consiste esencialmente en la predicción de bloques INTRA seguida de la cuantificación descrita en III.3.2.4 y la utilización de diferentes órdenes de exploración y de un cuadro VLC diferente para la decodificación de entropía de los coeficientes cuantificados.

La codificación de los bloques INTRA se implementa escogiendo uno de los tres modos predictivos descritos en H.263. Obsérvese que H.263 utiliza los coeficientes DCT *reconstruidos* para realizar la predicción de bloques INTER, mientras que los coeficientes DCT *originales* se utilizan en el decodificador para decidir el modo de predicción. Este modelo de prueba describe la decisión del modo realizada en el codificador en base a los coeficientes DCT originales.

Los bloques de los coeficientes DCT utilizados durante la predicción se denominan $A(u, v)$, $B(u, v)$ y $C(u, v)$, siendo u y v los índices de las filas y columnas, respectivamente. $C(u, v)$ representa los coeficientes DCT del bloque a codificar, $A(u, v)$ representa el bloque de los coeficientes DCT inmediatamente superiores a $C(u, v)$ y $B(u, v)$ representa el bloque de coeficientes DCT inmediatamente a la izquierda de $C(u, v)$. La posibilidad de utilizar los valores de coeficientes reconstruidos para los bloques A y B en la predicción de los valores de los coeficientes del bloque C depende de que los bloques A y B se encuentren en el mismo segmento de imagen de vídeo que el bloque C. Los segmentos de imagen se definen en el anexo R de H.263. $E_i(u, v)$ representa el error previsto para el modo INTRA siendo $i = 0, 1, 2$. Los errores de predicción se calculan para los tres modos de codificación de la manera siguiente:

modo 0: sólo predicción DC.

Si (tanto el bloque A como el bloque B están codificados INTRA y se encuentran en el mismo segmento de imagen que el bloque C)

```
{
  E0(0,0) = C(0,0) - ( A(0,0) + B(0,0) )//2
}
```

de lo contrario

```
{
  Si (el bloque A está codificado intra y se encuentra en el mismo segmento
  de imagen que el bloque C)
```

```
{
  E0(0,0) = C(0,0) - A(0,0)
}
```

de lo contrario

```
{
  Si (el bloque B está codificado intra y se encuentra en el mismo segmento
  de imagen que el bloque C)
```

```
{
  E0(0,0) = C(0,0) - B(0,0)
}
```

de lo contrario

```
{
  E0(0,0) = C(0,0) - 1024
}
```

```
}
```

$E0(u,v) = C(u,v) \quad u \neq 0, v \neq 0, u = 0..7, v = 0..7.$

modo 1: predicción de DC y AC del bloque anterior.

Si (el bloque A está codificado intra y se encuentra en el mismo segmento de imagen que el bloque C)

```
{
  E1(0,v) = C(0,v) - A(0,v)      v = 0..7, y
  E1(u,v) = C(u,v)              u = 1..7, v = 0..7.
}
```

de lo contrario

```
{
  E1(0,0) = C(0,0) - 1024
  E1(u,v) = C(u,v)      (u,v) != (0,0), u = 0,..7, v = 0,..7
}
```

modo 2: predicción de DC y AC desde el bloque hacia la izquierda.

Si (el bloque B está codificado intra y se encuentra en el mismo segmento de imagen que el bloque C)

```
{
  E2(0,v) = C(u,0) - A(u,0)      u = 0..7, y
  E2(u,v) = C(u,v)              v = 1..7, u = 0..7.
}
```

de lo contrario

```
{
  E2(0,0) = C(0,0) - 1024
  E2(u,v) = C(u,v)      (u,v) != (0,0), u = 0,..7, v = 0,..7
}
```

La selección del modo de predicción para la codificación INTRA avanzada se efectúa calculando la suma de valores absolutos del error de predicción, $SAD_{\text{modo } i}$, para los cuatro bloques de luminancia del macrobloque y seleccionando el modo que tenga el valor mínimo.

$$SAD_{\text{modo } i} = \sum_b \left[|E_i(0,0)| + 32 \sum_u |E_i(u,0)| + 32 \sum_v |E_i(0,v)| \right]$$

donde:

$$i = 0 \dots 3,$$

$$b = 0 \dots 3,$$

$$u, v = 1 \dots 7.$$

Una vez seleccionado el modo apropiado, se realiza la cuantificación. Los bloques se cuantifican de acuerdo con lo descrito en III.3.2.4.

III.3.5 Modo cuantificación modificada (anexo T)

La cuantificación modificada del anexo T reduce sobremanera ciertas perturbaciones de color (especialmente a velocidades binarias bajas) y aumenta el intervalo de los coeficientes de luminancia. Además, el anexo T permite que el codificador asigne al tamaño del paso cuantificador cualquier valor de granularidad del macrobloque que pueda mejorar la calidad de funcionamiento de los algoritmos de control de velocidad. El anexo T es obligatorio en todos los supuestos de aplicación y es muy recomendable para los diseños de productos de baja velocidad binaria.

III.4 Algoritmos peculiares de determinados supuestos de aplicación

Esta cláusula trata de los algoritmos que se utilizan en supuestos de aplicación específicos.

III.4.1 Decisión del modo

La Recomendación UIT-T H.263 contempla varios sistemas de codificación de macrobloques, tales como el modo INTRA (que codifica coeficientes DCT no predictivos), el modo INTER (de codificación predictiva con el vector de movimiento 1) y el INTER4V (codificación predictiva con 4 vectores de movimiento). La selección del modo adecuado constituye una de las funcionalidades clave del codificador y la bondad de la decisión repercute enormemente en la calidad de funcionamiento del codificador. Las subcláusulas siguientes describen los algoritmos de alta calidad y de baja complejidad desarrollados para el entorno libre de errores y para el propenso a errores. La primera descripción corresponde a un mecanismo de ejecución de la actualización INTRA obligatoria.

III.4.1.1 Patrón de renovación y actualización de los macrobloques INTRA

La Recomendación UIT-T H.263 establece la obligación de que cada macrobloque se codifique en modo INTRA una vez de cada 132 como mínimo, durante la transmisión de los coeficientes (a menos que se utilice el anexo W IDCT). A fin de evitar grandes ráfagas de macrobloques INTRA durante periodos breves, se utiliza un patrón sencillo de actualización del macrobloque para inicializar aleatoriamente los contadores de actualización de los macrobloques a un valor comprendido en el intervalo [0,132]. Para inicializar el patrón aleatorio debe utilizarse el generador pseudoaleatorio del anexo A a la Recomendación UIT-T H.263. Este proceso de decisión del modo prevalece sobre el modo de codificación correspondiente a cualquier otro proceso de decisión del modo. El algoritmo en cuestión se describe a continuación:

Actualización_MB_intra [posx][posy]: se incrementa el contador en uno cada vez que se envía la información del coeficiente para el macrobloque situado en (posx, posy), es decir MB[posx][posy].

Frecuencia_Renovación_MB_INTRA: valor constante (132) en entornos libres de error. En entornos propensos a error, Frecuencia_Renovación_MB_INTRA puede ser una variable entera (adaptada a la tasa de errores), cuyo valor se encuentre en (1, 132).

El modo INTRA para un determinado macrobloque se escoge así:

```
Inicializar tras una imagen I: Actualización_intra_MB[posx][posy] =
aleatorizador_del_anexo_A (0, Frecuencia_Renovación_MB_INTRA)
Mientras (haya más imágenes para codificar que no sean I)
  Si ((actualización_intra_MB[posx][posy]
    == FRECUENCIA_RENOVACIÓN_MB_INTRA) &&
    (MB[posx][posy] contiene el coeficiente))
  {
    Codificar MB[posx][posy] en modo INTRA;
  }
de lo contrario si (MB[posx][posy] contiene coeficiente)
{
  ++Actualización_intra_MB[posx][posy];
}
```

Puede obtenerse más información en [Q15-E-15] y [Q15-E-37].

III.4.1.2 Decisión del modo de baja complejidad para entornos libres de errores

La figura III.3 presenta un esquema del algoritmo de decisión del modo de baja complejidad, que interactúa con las etapas de búsqueda del vector de movimiento de baja y media calidad descritas en III.3.1.2 y III.3.1.3 respectivamente.

Una vez realizada la estimación del movimiento de píxel entero, el codificador selecciona el modo de predicción INTRA o el INTER. Para adoptar la decisión INTRA/INTER se calculan los siguientes parámetros:

$$\mu_{MB} = \frac{1}{256} \sum_{i=1}^{16} \sum_{j=1}^{16} X(i, j)$$

$$A = \sum_{i=1}^{16} \sum_{j=1}^{16} |X(i, j) - \mu_{MB}|$$

Se escogerá el modo INTRA cuando:

$$A < (SAD(u, v) - 500)$$

Si se escoge el modo INTRA ya no es necesario ejecutar pasos adicionales de estimación de movimiento o de decisión del modo. Si se utiliza el control de velocidad, los coeficientes DCT de los bloques se cuantifican con el parámetro de cuantificación determinado por el control de velocidad. Las operaciones correspondientes al modo de codificación INTRA avanzada se realizan de acuerdo con lo definido en III.3.4. Se calcula el patrón de bloque codificado (CBP, *coded block pattern*) y se genera el macrobloque correspondiente a este tren binario.

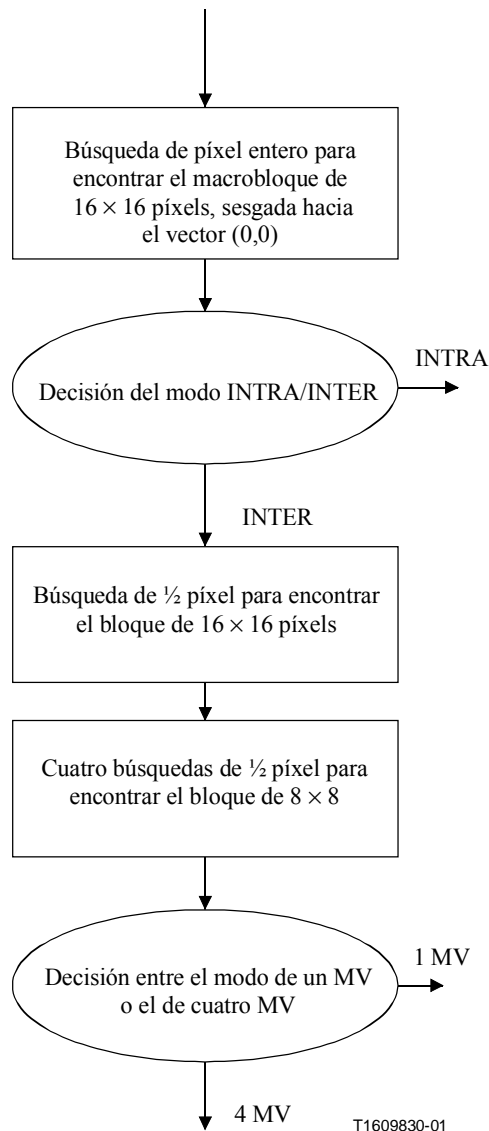


Figura III.3/H.263 – Diagrama de bloques del proceso de decisión del modo de baja complejidad

Si se escoge el modo INTER continúa la búsqueda con la afinación a medio píxel en torno al vector de movimiento de píxel entero de 16×16 .

Si se admiten los vectores de movimiento de 8×8 , es decir si están vigentes los anexos F o J, también continúa la búsqueda de movimiento de medio píxel en torno a los vectores de movimiento de píxel entero de 8×8 . Acto seguido la decisión del modo determina si se ha de utilizar la predicción INTER o la INTER4V.

La SAD correspondiente al vector 16×16 de medio píxel de precisión (incluida la resta de 100 si el vector es $(0,0)$) viene dada por:

$$SAD_{16}(u, v)$$

La SAD agregada correspondiente a los cuatro vectores 8×8 de medio píxel de precisión viene dada por:

$$SAD_{8 \times 8} = \sum_{k=1}^4 SAD_{8,k}(u_k, v_k)$$

el modo INTER4V se selecciona cuando:

$$SAD_{8 \times 8} < SAD_{16} - 200$$

de lo contrario se selecciona el modo INTER.

III.4.1.3 Decisión del modo alta calidad correspondiente a entornos libres de errores

En primer lugar se lleva a cabo la estimación del movimiento de alta calidad correspondiente a los modos INTER e INTER4V, como se describe en III.3.1.4. Una vez determinados estos vectores de movimiento, se calculan los costes totales de distorsión de velocidad para todos los modos de codificación contemplados.

Independientemente, el codificador selecciona el modo de codificación de macrobloques que minimiza

$$J = D + \lambda_{\text{modo}} \times R$$

La distorsión, D , se define como la suma de los cuadrados de las diferencias (SSD, *sum of squared difference*) entre los coeficientes de las componentes de luminancia y crominancia del macrobloque objetivo y los coeficientes de los componentes cuantificados de luminancia y crominancia correspondientes al modo de codificación en cuestión. R se define como la velocidad de codificación del macrobloque objetivo incluida toda la información de control, movimiento y textura, correspondiente al modo de codificación en cuestión.

El parámetro λ viene dado por:

$$\lambda_{\text{modo}} = 0,85 \cdot QP^2$$

siendo QP el parámetro de cuantificación del macrobloque [Q15-D-13].

Los modos de codificación probados explícitamente son los siguientes SKIPPED, INTRA, INTER, e INTER4V. El modo INTER utiliza el mejor vector de movimiento de 16×16 y el INTER4V utiliza los mejores vectores de movimiento 8×8 seleccionados mediante un algoritmo de estimación de movimiento de alta calidad. Si la aplicación es de velocidad binaria fija, se modifica la relación de acuerdo con lo descrito en III.4.2.3.

Cuando rige el anexo F, no se considera la ventana de compensación de movimiento de bloque solapado en la adopción de decisiones de modos de codificación, aplicándose solamente una vez determinados todos los vectores de movimiento y una vez adoptadas todas las decisiones de modo para la imagen.

III.4.1.4 Decisión de modo de baja complejidad correspondiente a entornos propensos a error

El algoritmo de decisión del modo de baja complejidad para entornos propensos a error utiliza el mecanismo descrito en III.4.1.2, con una adición. La frecuencia de actualización del macrobloque INTRA se incrementa al adoptar para el parámetro Frecuencia_Renovación_MB_INTRA de III.4.1.1 un valor inferior a 132. Las subcláusulas III.2.3.2 y III.2.4.1 contienen directrices para la selección del valor de esta variable.

III.4.1.5 Decisión del modo de alta calidad correspondiente a entornos propensos a error

El algoritmo de decisión del modo de alta calidad correspondiente a los entornos propensos a error es semejante al descrito en III.4.1.3. El codificador selecciona para cada macrobloque, independientemente de los demás, el modo de codificación que minimiza:

$$J = (1 - p)D_1 + pD_2 + \lambda_{\text{modo}}R$$

siendo λ_{modo} la definida en III.4.1.3. Para computar este coste, el codificador simula el método de ocultación de errores empleado en el decodificador, debiendo obtener la probabilidad p de que un

cierto macrobloque se pierda durante la transmisión. De este modo la distorsión procede de dos fuentes: D_1 , que es la distorsión imputable al macrobloque recibido, y D_2 , que es la distorsión imputable al macrobloque perdido y oculto.

Para un determinado MB, se calculan dos distorsiones en todos los modos de codificación considerados: la distorsión de codificación D_1 y la distorsión de ocultación D_2 . D_1 se pondera con la probabilidad $(1 - p)$ de que se reciba este MB sin errores, y D_2 se pondera con la probabilidad p de que el mismo MB se pierda y se oculte. D_2 depende del método de ocultación de errores empleado y es constante para todos los modos de codificación considerados (no repercutiendo por consiguiente en la selección del modo mediante la anterior ecuación). D_1 dependerá del modo de codificación considerado. Para el modo INTRA la distorsión coincide con la del caso libre de errores. Para los modos SKIP e INTER, D_1 es la distorsión de cuantificación más la distorsión de ocultación del macrobloque de la que se predice, más exactamente,

$$D_1 = D_q + pD_2(v, n - 1)$$

siendo $D_2(v, n - 1)$ la distorsión de ocultación del área de la imagen anterior, $(n - 1)$, en la posición espacial actual del macrobloque desplazado por el vector de movimiento v , poniéndose v a *cero* en el modo SKIP y al vector de movimiento del macrobloque considerado en el modo INTER.

D_q se define como la suma de los cuadrados de las diferencias entre los valores de los píxels adecuadamente reconstruidos del macrobloque codificado y los valores iniciales de los píxels, y D_2 se define como la suma de los cuadrados de las diferencias entre los valores de los píxels adecuadamente reconstruidos del macrobloque codificado y los valores de los píxels ocultos. R se define como la velocidad de codificación del macrobloque objetivo incluida toda la información de control, movimiento y textura, correspondiente al modo de codificación en cuestión.

III.4.2 Control de velocidad

Este algoritmo de control de velocidad es aplicable a todos los supuestos de aplicación de velocidad binaria fija. Consiste en una capa de trama, en la que se selecciona una velocidad binaria objetivo para la imagen actual, y una capa de macrobloque en la que se adapta el parámetro de cuantificación (QP, *quantization parameter*) para obtener dicho objetivo. El desarrollo de este algoritmo se describe en [Q15-A-20]. Este algoritmo de control de velocidad, denominado frecuentemente control de velocidad TMN8, es el único algoritmo de control de velocidad que se utiliza en los supuestos de aplicación.

Inicialmente, el número de bits de la memoria intermedia W se pone a cero, $W = 0$, y los parámetros K_{prev} y C_{prev} se inicializan haciendo $K_{prev} = 0,5$ y $C_{prev} = 0$. La primera imagen se codifica INTRA utilizando un valor fijo de QP para todos los macrobloques.

III.4.2.1 Control de velocidad a nivel de la trama

En la descripción de este algoritmo se utilizan las siguientes definiciones:

- B' Número de bits ocupados por la anterior imagen codificada.
- R Velocidad binaria objetivo en bits por segundo (por ejemplo, 10 000 bit/s, 24 000 bit/s, etc.).
- G Frecuencia de trama de la secuencia de vídeo original en imágenes por segundo (por ejemplo 30).
- F Frecuencia de la trama objetivo en imágenes por segundo (por ejemplo 7,5, 10, etc.). G/F debe ser entero.
- M Umbral de salto de trama. Por defecto se pone $M = R/F$. (M/R es el máximo retardo de la memoria intermedia.)
- A El objetivo de retardo de la memoria intermedia es AM . Por defecto, se pone $A = 0,1$.

El número de bits de la memoria intermedia del codificador es $W = \max (W + B' - R/F, 0)$. El parámetro de salto se pone a 1, salto = 1.

```
Mientras W > M
{
    W = max (W - R/F, 0)
    salto++
}
```

Codificación del salto de las siguientes (salto*(G/F) - 1) imágenes de la secuencia de vídeo original. El objetivo del número de bits por imagen es $B = (R/F) - \Delta$, siendo:

$$\Delta = \begin{cases} W, & W > A \cdot M \\ \frac{W}{F}, & \\ W - A \cdot M, & \text{En caso contrario} \end{cases}$$

En las aplicaciones con frecuencia de trama fija, el salto es siempre igual a 1. El salto de trama es constante e igual a $(G/F) - 1$. Además, cuando se calcula el objetivo de número de bits por imagen, A debe ser 0,5.

III.4.2.2 Control de velocidad a nivel del macrobloque

Paso 1: Inicialización

Se supone completada la estimación del vector de movimiento.

σ_k^2 se define como la varianza de los valores de luminancia y crominancia del macrobloque k-ésimo.

Si el macrobloque k-ésimo es del tipo I (intra), hacer $\sigma_k^2 = \sigma_k^2 / 3$.

Sean $i = 1$ y $j = 0$,

$\tilde{B}_1 = B$ objetivo del número de bits definido en III.4.2.1,

$N_1 = N$ número de macrobloques de una imagen,

$K = K_1 = K_{\text{prev}}$ valores iniciales de los parámetros del modelo, y

$C = C_1 = C_{\text{prev}}$ valores iniciales de los parámetros del modelo.

$$S_1 = \sum_{k=1}^N \alpha_k \sigma_k, \text{ siendo } \alpha_k = \begin{cases} 2 \frac{B}{16^2 N} (1 - \sigma_k) + \sigma_k, & \frac{B}{16^2 N} < 0.5, \\ 1, & \text{En los demás casos.} \end{cases}$$

Paso 2: Calcular la Q optimizada para el macrobloque i-ésimo

Si $L = (\tilde{B}_i - 16^2 N_i C) \leq 0$ (agotando los bits), hacer $Q_i^* = 62$.

En caso contrario, calcular:

$$Q_i^* = \sqrt{\frac{16^2 K}{L} \frac{\sigma_i}{\alpha_i} S_i}$$

Paso 3: Calcular QP y codificar el macrobloque

$QP = (Q_i^* / 2)$ redondeado al entero más próximo del conjunto 1, 2, ..., 31.

$DQUANT = QP - QP_{\text{prev}}$.

Si $DQUANT > 2$, hacer $DQUANT = 2$. Si $DQUANT < -2$, hacer $DQUANT = -2$;

Hacer $QP = QP_{prev} + DQUANT$.

Codificar DCT el macrobloque con el parámetro de cuantificación QP , y hacer $QP_{prev} = QP$.

Paso 4: Actualizar los contadores

Siendo B'_i el número de bits utilizados para codificar el macrobloque i -ésimo calcular:

$$\tilde{B}_{i+1} = \tilde{B}_i - B'_i, S_{i+1} = S_i - \alpha_i \sigma_i, \text{ y } N_{i+1} = N_i - 1$$

Paso 5: Actualizar los parámetros K y C del modelo

Los parámetros del modelo medidos para el macrobloque i -ésimo son los siguientes:

$$\hat{K} = \frac{B'_{LC,i} (2QP)^2}{16^2 \sigma_i^2} \text{ y } \hat{C} = \frac{B'_i - B'_{LC,i}}{16^2}$$

siendo $B'_{LC,i}$ el número de bits utilizados para la luminancia y crominancia del macrobloque.

A continuación, se miden los \hat{K} y \hat{C} medios de la imagen calculados hasta el momento.

si ($\hat{K} > 0$ y $\hat{K} \leq \pi \log_2 e$), hacer $j = j + 1$ y calcular $\tilde{K}_j = \tilde{K}_{j-1}(j-1)/j + \hat{K}/j$.

Calcular $\tilde{C}_i = \tilde{C}_{i-1}(i-1)/i + \hat{C}/i$.

Finalmente, las actualizaciones son la media ponderada de las estimaciones iniciales, K_1 , C_1 , y de su media actual:

$$K = \tilde{K}_j(i/n) + K_1(N-i)/N, \quad C = \tilde{C}_i(i/n) + C_1(N-i)/N.$$

Paso 6:

Si $i = N$, parar (se han codificado todos los macrobloques).

Hacer $K_{prev} = K$ y $C_{prev} = C$.

De lo contrario, hacer $i = i + 1$, e ir al Paso 2.

III.4.2.3 Los modelos de alta calidad y el control de velocidad

En las minimizaciones por Lagrangiana como las descritas en los algoritmos de estimación de movimiento y de decisión del modo de alta calidad, de III.3.1.4 y III.4.1.3, respectivamente, el parámetro de Lagrange debe permanecer constante a lo largo de toda la imagen. Esto se justifica porque en la situación óptima, todas las regiones de la imagen deben funcionar en un punto de pendiente constante de las curvas de distorsión de velocidad [SG88]. En teoría, este punto de funcionamiento óptimo se determina mediante la exploración de todos los valores posibles de λ . No obstante, en este modelo de prueba, se define una relación sencilla entre λ y el parámetro de cuantificación que produce una buena aproximación al valor óptimo de λ [Q15-D-13].

Como el algoritmo de control de velocidad descrito anteriormente puede actualizar el parámetro de cuantificación en cada macrobloque codificado, las relaciones entre λ y el parámetro de cuantificación deben modificarse ligeramente. Se ha demostrado que, para emplear los modelos de alta calidad en supuesto de velocidad binaria fija, los parámetros de Lagrange deben calcularse de la siguiente manera:

$$\lambda_{movimiento} = 0,92 \cdot \overline{QP}_{prev} \text{ y}$$

$$\lambda_{modo} = 0,85 \cdot \overline{QP^2}_{prev}$$

respectivamente, siendo \overline{QP}_{prev} el valor promediado a lo largo de todos los macrobloques del parámetro de cuantificación de la anterior imagen codificada del mismo tipo que la actual.

Los algoritmos de estimación de movimiento y de decisión del modo de alta calidad determinan a continuación vectores de movimiento y modos de codificación de macrobloques adecuados, al igual que los supuestos de velocidad binaria variable, es decir como si no se utilizara ningún algoritmo de control de velocidad. Una vez determinados los vectores de movimiento y los modos óptimos, el algoritmo de control de velocidad puede utilizarse para codificar la imagen en base a los vectores de movimiento y a los modos de codificación ya determinados. En el caso particular de que el algoritmo de decisión del modo haya seleccionado el modo *SKIPPED* para el macrobloque objetivo, no se permite que el algoritmo de control de velocidad actualice el parámetro de cuantificación.

III.4.2.4 Selección de la imagen de referencia mejorada (anexo U)

El modo selección de imagen de referencia mejorada (ERPS, *enhanced reference picture selection*) (anexo U) activa la predicción multi-imagen compensada en movimiento (MCP, *motion-compensated prediction*). Cuando se está en el modo ERPS, las M ($M \geq 1$) imágenes anteriores decodificadas se introducen en una memoria intermedia tanto en el codificador como en el decodificador. A diferencia de la MCP de imagen sencilla, la MCP de múltiple imagen puede explotar las dependencias estadísticas a largo plazo de las secuencias de vídeo. Son ejemplos representativos de estas dependencias los cortes de secuencia, los trasfondos interiores y exteriores, los contenidos que aparecen y desaparecen de la imagen, etc.

III.4.2.4.1 Selección del número de imágenes de referencia

En los sistemas reales el número M de imágenes de referencia que el decodificador puede acomodar se negocia o se define de cualquier otro modo por medios externos. Las mejoras en la distorsión de velocidad con respecto a la MCP de imagen sencilla dependen del número de imágenes de referencia y de la manera en que se utilizan. Normalmente, cuantas más imágenes de referencia haya mejor será la calidad de funcionamiento de la codificación. No obstante, un número de imágenes de referencia mayor provoca asimismo un aumento de la complejidad y de los requisitos de memoria. Cuando se promedian los resultados de varias secuencias de prueba, resulta que las mejoras de la PSNR a velocidad binaria fija suelen ser proporcionales al logaritmo ($\log(M)$) del número de imágenes de referencia. Por consiguiente, a efectos del modelo de prueba, se utilizan diez imágenes de referencia para obtener un buen compromiso entre el rendimiento de codificación y la complejidad. Aunque un codificador pueda alcanzar una calidad de funcionamiento mejorada por medio de una selección optimizada de las imágenes de referencia a almacenar (e incluso tal vez una selección optimizada de las regiones subimagen a almacenar), el modelo de prueba se limita a almacenar y utilizar las imágenes de referencia según su orden de llegada (FIFO, *first-in, first-out*).

III.4.2.4.2 Estimación del movimiento

La estimación del movimiento se desarrolla utilizando las imágenes de referencia disponibles en la memoria intermedia multi-imagen (hasta un máximo de M , que en el modelo de prueba tiene el valor 10). Para determinar el vector de movimiento óptimo \mathbf{m}_k para el k -ésimo macrobloque o bloque, ha de minimizarse la siguiente función Lagrangiana de coste:

$$\mathbf{m}_k = \underset{\mathbf{m} \in \text{SR}}{\operatorname{argmin}} D(\mathbf{s}, \mathbf{c}(\mathbf{m})) + \lambda_{\text{movimiento}} R(\mathbf{m} - \mathbf{p})$$

siendo SR el intervalo de búsqueda que suele abarcar el conjunto de posiciones de píxel entero $[-32 \dots 31] \times [-32 \dots 31] \times [0 \dots M - 1]$ en las direcciones horizontal, vertical y temporal.

NOTA – En la mayor parte de los experimentos realizados durante la fase de propuesta de las pruebas del modo ERPS se utilizó en realidad un intervalo de búsqueda de muestras enteras de ± 16 ; no obstante, aquí se especifica un intervalo más amplio debido a que el diseño del modelo de prueba tiene un intervalo superior.

La distorsión $D(s, c(m))$ se especifica como SAD y el término $R(m - p)$ determina la velocidad incluido el desplazamiento espacial m_x, m_y y el parámetro de referencia de imagen m_t dado el predictor $p = (p_x, p_y, 0)$. Tras la determinación del vector de movimiento de píxel entero del macrobloque o bloque, se aplican los pasos de codificación restantes especificados en el modo alta calidad de III.4.2.3. Al parámetro Lagrange $\lambda_{\text{movimiento}}$ se le da el valor $\lambda_{\text{movimiento}} = 0,92 \text{ QP}$.

III.4.2.5 Utilización de rebanadas de datos particionadas (anexo V)

El modo rebanada de datos particionada se utiliza en el modo V3 del supuesto de aplicación muy propenso a errores binarios. Se le considera capaz de mejorar asimismo la calidad de funcionamiento en entornos de pérdidas de paquetes, con tal de que los sistemas de empaquetado utilicen como mínimo un paquete para cada partición. Si se reorganiza la sintaxis formando rebanadas con tres particiones, una para cada cabecera, vector de movimiento y coeficientes DCT, se reduce la propagación de errores entre particiones y se activa la utilización de RVLC. Al contrario que en otras partes de este modelo de prueba, esta cláusula presenta las descripciones del algoritmo de funcionamiento del codificador y del decodificador.

III.4.2.5.1 Funcionamiento del codificador

La codificación fuente de los datos permanece inalterada, sólo cambia la sintaxis de las rebanadas con respecto a lo especificado en el anexo K. Con objeto de formatear el vídeo codificado en rebanadas, se utiliza una estructura de paquetes de longitud pseudofija. El codificador escoge un umbral de longitud de rebanada t , que debe seleccionarse en función de las características de error de la red y de la tara. Una vez escogido t , el codificador codifica un número entero de MB de modo que la longitud de la rebanada resultante sea inferior a t . Teóricamente, t haría que las rebanadas tuvieran como máximo de uno a dos errores para aprovechar al máximo la recuperación del RVLC, por lo tanto un buen límite inferior para el tamaño del paquete es $0,5 * 1/\text{BER}$ bits, siendo $\text{BER} = \text{frecuencia media de errores en los bits del canal}$. No obstante, esto no es siempre posible debido a la tara que introducen las rebanadas muy cortas. Considerando todo esto, la implementación del modelo de prueba del anexo V ha adoptado para t el valor de 700 bits, ya que este número ha proporcionado buenos resultados para el canal WCDMA de 64 kbit/s y $10E - 3 \text{ BER}$ (que es el canal más desfavorable de los contemplados en las Condiciones Comunes [Q15-I-60]).

III.4.2.5.2 Método de decodificación

Esta subcláusula describe el método utilizado para decodificar el tren binario de vídeo, con las técnicas de detección de errores y ocultación utilizadas. La descripción informal de los algoritmos prevalece sobre las definiciones más genéricas de III.5.3. La ocultación de errores suele ser independiente del método aquí descrito y se efectúa por tanto con arreglo al método descrito en III.5.4.

El método de decodificación de datos de vídeo real es idéntico al de codificación con la única diferencia de la recuperación de datos y manejo de errores propios de la nueva estructura de datos particionados.

- 1) Explorar el tren binario en busca de un PSC. Cuando se recibe un PSC se supone que le sigue inmediatamente una cabecera de imagen.
- 2) Decodificar la cabecera de imagen. Coincide exactamente con el caso de no utilización del anexo V. Si se detecta un error en la cabecera de imagen, se utiliza la información de la última cabecera de imagen libre de errores conocida o del duplicado del anexo W, de haberlo.
- 3) Explorar el siguiente SSC. El siguiente SSC debe seguir inmediatamente a la cabecera de la película y al SSTUF. De lo contrario, se supone que se trata de un error y se ignoran todos los bits entre la cabecera de imagen y este SSC.

NOTA – Dado que el SSC es un prefijo del PSC, se comprobará si se trata efectivamente de un PSC. En caso afirmativo, dejará de ejecutarse, lógicamente, la siguiente iteración.

- 4) Repetir lo siguiente hasta la detección del siguiente PSC:
 - a) Decodificar SEPB1 y SSBI.
 - b) Justo a continuación del SSBI hay un MBA. Pueden presentarse dos casos:
 - i) Si esta rebanada sigue inmediatamente a un PSC y a una cabecera de imagen, el primer índice MB debería ser 0. De no serlo, se pondría a 0 y se continuaría decodificando la rebanada.
 - ii) En caso contrario, el primer índice MB debería basarse en la rebanada inmediatamente anterior. Pueden darse dos casos:
 - no hay errores en la rebanada inmediatamente anterior: se otorga a MBA el valor calculado a partir de la rebanada inmediatamente anterior;
 - se ha perdido o hay errores en la rebanada inmediatamente anterior: se da a MBA el valor leído.
 - c) Decodificar SEPB2, SQUANT, SWI, SEPB3 y GFID
 - d) Buscar la siguiente marca de cabecera (HM, *header marker*).
 - e) Decodificar la partición de la cabecera de datos (HD, *header data*). La decodificación sólo se efectúa en sentido directo. Si se detectan errores, se ignora toda la rebanada hasta el siguiente PSC o SSC.
- 5) Buscar la marca de movimiento (MVM). Si se encuentra antes un PSC o un SSC, se considera error y se ignora la rebanada.
- 6) Decodificar la partición de vectores de movimiento (MVD y LMVV). Esto se hace tanto en sentido directo como en inverso. Conservar todos los vectores de movimiento decodificados sin error tanto en sentido directo como en inverso.
- 7) Buscar el siguiente PSC o SSC.
- 8) Decodificar la partición de datos de coeficientes. Como está codificada con VLC y no con RVLC, sólo podemos decodificar en sentido directo. Si se detectan errores en algún punto, todos los coeficientes DCT se ponen a 0, lo que equivale a dar por perdida la información residual de errores.

III.4.2.6 Utilización del algoritmo de cabecera anterior del anexo W

A diferencia de algoritmo recomendado en 5.1.4.3 de la versión 2 de H.263, el bit de tipo de redondeo de PLUSPTYPE se pone a "0" independientemente del tipo de imagen [Q15-I-26]. Esto contribuye a mantener un valor constante de GFID, lo que permite al decodificador recuperar una cabecera de imagen perdida de la siguiente manera: si el GFID de la cabecera del primer GOB decodificable coincide con el de la imagen anterior, los datos de la imagen se decodifican utilizando la cabecera de imagen anterior. De lo contrario, si el GFID es distinto del de la imagen anterior, se inserta una copia de la cabecera de imagen en la información de mejora complementaria de la imagen siguiente, definida en W.6.3.6 de H.263 y el decodificador utiliza esta copia para recuperar la cabecera de imagen perdida. Véase [Q15-J-62] y las referencias adjuntas.

III.5 Posprocesamiento del decodificador

Esta cláusula trata del funcionamiento del decodificador de vídeo en entornos propensos a error, así como de los métodos de posprocesamiento tales como los algoritmos de ocultación de errores y el posfiltrado. Aunque el posfiltrado es útil en entornos sin errores y en los propensos a error, los demás mecanismos aquí descritos son aplicables exclusivamente a los entornos propensos a error.

III.5.1 Posfiltrado de la sobreoscilación

El posfiltrado de la sobreoscilación descrito en esta subcláusula debe utilizarse siempre que se utilice el anexo J. De lo contrario, deberá emplearse el posfiltrado de desbloqueo descrito en III.5.2. En lo que respecta a la definición de las funciones y la notación que esta subcláusula utiliza, véase la definición de filtro de bucle que figura en el anexo J de H.263.

Se describirá la versión unidimensional del filtro. Para obtener un efecto bidimensional, el filtro se utiliza primero en sentido horizontal y después en sentido vertical. Los píxels A, B, C, D, E, F, G, (H) se alinean horizontalmente o verticalmente. El filtro producirá un nuevo valor, D1, para D:

$D1 = D + \text{Filtro}((A + B + C + E + F + G - 6D)/8, \text{Profundidad } 1)$ filtrando en la primera dirección.

$D1 = D + \text{Filtro}((A + B + C + E + F + G - 6D)/8, \text{Profundidad } 2)$ filtrando en la segunda dirección.

A diferencia del filtrado del anexo J, el posfiltrado se aplica a todos los píxels de la imagen. Cuando se efectúa el filtrado de los límites de la imagen, deben repetirse los píxels de los bordes. Es posible que profundidad 1 y profundidad 2 sean distintas para obtener una mejor adaptación a QUANT de la profundidad total del filtrado. El cuadro III.1 muestra la relación entre profundidad 1, profundidad 2 y QUANT. Profundidad 1 y profundidad 2 pueden relacionarse con el QUANT del macrobloque al que pertenece D o con un valor de QUANT promediado sobre ciertas partes de la imagen o sobre toda ella.

Para obtener la suma de 7 píxels (A + B + C + D + E + F + G) puede utilizarse una técnica de ventana deslizante. De este modo, puede reducirse el número de operaciones para realizar el filtro.

Cuadro III.1/H.263 – TMN

QUANT	Profundidad	Profundidad 1	Profundidad 2	QUANT	Profundidad	Profundidad 1	Profundidad 2
1	1	1	1	17	8	3	3
2	1	1	1	18	8	3	3
3	2	1	1	19	8	3	3
4	2	1	1	20	9	3	3
5	3	1	1	21	9	3	3
6	3	2	1	22	9	3	3
7	4	2	1	23	10	3	3
8	4	2	2	24	10	4	3
9	4	2	2	25	10	4	3
10	5	2	2	26	11	4	3
11	5	3	2	27	11	4	3
12	6	3	2	28	11	4	3
13	6	3	2	29	12	4	3
14	7	3	2	30	12	4	3
15	7	3	3	31	12	4	3
16	7	3	3				

III.5.2 Posfiltrado de desbloqueo

Esta subcláusula describe un posfiltrado bidimensional regularizado destinado a reducir al mismo tiempo el bloqueo y las perturbaciones de sobreoscilación. Este filtrado es recomendable cuando no se utiliza el anexo J, a fin de evitar resultados excesivamente difuminados. El modelo de la degradación introducida en el proceso de cuantificación puede describirse como:

$$g(i, j) = f(i, j) + n(i, j)$$

siendo g, f, n la imagen reconstruida, la original y el ruido de cuantificación en el dominio espacial, e (i, j) las coordenadas vertical y horizontal de la imagen de vídeo. Para minimizar el ruido de cuantificación se introduce una métrica cuatridimensional del alisado, M , que se define del siguiente modo:

$$M(f(i, j)) = M_L(f(i, j)) + M_R(f(i, j)) + M_U(f(i, j)) + M_D(f(i, j))$$

$$M_L(f(i, j)) = (1 - \alpha_L(f(i, j))) [f(i, j) - g(i, j)]^2 + \alpha_L(f(i, j)) [f(i, j) - f(i, j-1)]^2$$

$$M_R(f(i, j)) = (1 - \alpha_R(f(i, j))) [f(i, j) - g(i, j)]^2 + \alpha_R(f(i, j)) [f(i, j) - f(i, j+1)]^2$$

$$M_U(f(i, j)) = (1 - \alpha_U(f(i, j))) [f(i, j) - g(i, j)]^2 + \alpha_U(f(i, j)) [f(i, j) - f(i-1, j)]^2$$

$$M_D(f(i, j)) = (1 - \alpha_D(f(i, j))) [f(i, j) - g(i, j)]^2 + \alpha_D(f(i, j)) [f(i, j) - f(i+1, j)]^2$$

siendo α_i ($i = L, R, U, D$) los parámetros de regularización que controlan la contribución relativa entre fidelidad y alisado direccional. La figura III.4 muestra un ejemplo de la situación de estos píxels.

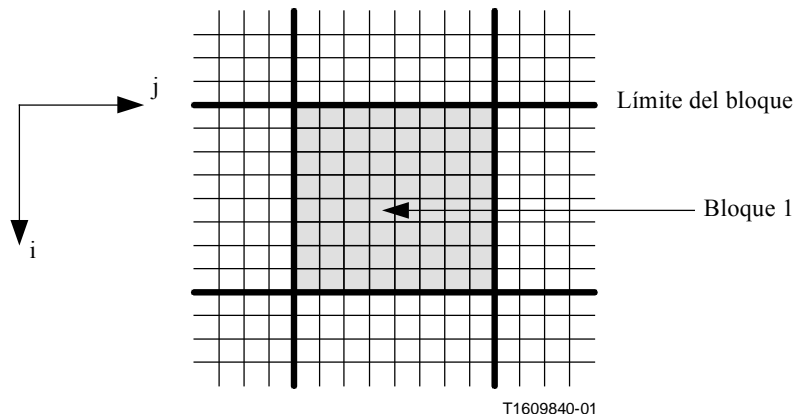


Figura III.4/H.263 – Ejemplo de situación de los píxels

El bit COD se emplea para estimar la correlación del macrobloque entre la trama actual y la anterior imagen codificada. Por consiguiente, se utilizarán distintos filtros en función del COD, lo que reducirá la complejidad. A continuación se describe el filtrado cuando COD es 0 (codificado) y cuando es 1 (no codificado).

Condición 1: el bloque 1 pertenece a un macrobloque codificado (COD = 0)

Si el bloque 1 de la figura III.4 (zona gris) pertenece a un macrobloque codificado, se aplicará el siguiente filtro a todos sus píxels. Si el píxel $g(i, j)$ pertenece al bloque 1, $f(i, j)$ se obtiene minimizando la siguiente función:

$$f(i, j) = \frac{(4 - \alpha_{TOT}(f(i, j)))g(i, j) + A}{4}$$

$$\alpha_{TOT}(f(i, j)) = \alpha_L(f(i, j)) + \alpha_R(f(i, j)) + \alpha_U(f(i, j)) + \alpha_D(f(i, j))$$

$$A = \alpha_L(f(i, j))g(i, j-1) + \alpha_R(f(i, j))g(i, j+1) + \alpha_U(f(i, j))g(i-1, j) + \alpha_D(f(i, j))g(i+1, j)$$

Los parámetros de regularización se calculan del siguiente modo:

$$\alpha_L(f(i, j)) = \frac{K_L QP_l^2}{[g(i, j) - g(i, j-1)]^2 + K_L QP_l^2}$$

$$\alpha_R(f(i, j)) = \frac{K_R QP_l^2}{[g(i, j) - g(i, j+1)]^2 + K_R QP_l^2}$$

$$\alpha_U(f(i, j)) = \frac{K_U QP_l^2}{[g(i, j) - g(i-1, j)]^2 + K_U QP_l^2}$$

$$\alpha_D(f(i, j)) = \frac{K_D QP_l^2}{[g(i, j) - g(i+1, j)]^2 + K_D QP_l^2}$$

siendo QP_l el valor del parámetro de cuantificación (tamaño del paso de cuantificación) del último macrobloque codificado. Como en los límites de los bloques se necesita un alisado más pronunciado y en el interior de los bloques un suavizado relativamente menos intenso, las constantes K_i ($i = L, R, U, D$) se determinan en función de la posición de $g(i, j)$ del siguiente modo:

$$K_L, K_U = \begin{cases} 9 & \text{si } j \bmod 8 = 0 \\ 1 & \text{en los demás casos} \end{cases}$$

$$K_R, K_D = \begin{cases} 9 & \text{si } j \bmod 8 = 7 \\ 1 & \text{en caso contrario} \end{cases}$$

Condición 2: el bloque 1 pertenece a un macrobloque no codificado (COD = 1)

Si el bloque 1 de la figura III.4 (zona sombreada) pertenece a un macrobloque no codificado, se aplica la siguiente regla a todos sus píxeles

$$f(i, j) = f_p(i, j)$$

siendo f_p los píxeles correspondientes de la anterior imagen recuperada.

Para facilitar la aplicación del algoritmo descrito puede utilizarse el siguiente fragmento de pseudocódigo:

```
f /* matriz bidimensional, trama recuperada */
g /* matriz bidimensional, trama reconstruida */
QP /* parámetro de cuantificación */
K_L /* constante izquierda del parámetro de regularización */
K_R /* constante derecha del parámetro de regularización */
K_U /* constante superior del parámetro de regularización */
K_D /* constante inferior del parámetro de regularización */
alpha_L /* parámetro de regularización a la izquierda */
alpha_R /* parámetro de regularización a la derecha */
alpha_U /* parámetro de regularización arriba */
alpha_D /* parámetro de regularización abajo */

QP=(QP*QP);
if(COD == 0)
{
    diff_L=(g(i,j)-g(i,j-1));
```

```

diff_R=(g(i,j)-g(i,j+1));
diff_U=(g(i,j)-g(i-1,j));
diff_D=(g(i,j)-g(i+1,j));

if(i mod 8 ==0){
    K_U=9;
    K_D=1;
} else if(i mod 8 == 7){
    K_U=1;
    K_D=9;
} else{
    K_U=1;
    K_D=1;
}

if(j mod 8 ==0){
    K_L=9;
    K_R=1;
} else if (j mod 8 == 7){
    K_L=1;
    K_R=9;
} else{
    K_L=1;
    K_R=1;
}

alpha_L=((K_L*QP)/(diff_L*diff_L+K_L*QP));
alpha_R=((K_R*QP)/(diff_R*diff_R+K_R*QP));
alpha_U=((K_U*QP)/(diff_U*diff_U+K_U*QP));
alpha_D=((K_D*QP)/(diff_D*diff_D+K_D*QP));

f(i,j)=(((4-alpha_L-alpha_R-alpha_U-alpha_D)*g(i,j)+alpha_L*g(i,j-
)+alpha_R*g(i,j+1)+alpha_U*g(i-1,j)+alpha_D*g(i+1,j))/4);
}

```

III.5.3 Detección de errores

Algunas partes de esta cláusula están basadas en el modelo de ocultación de errores TCON descrito en LBC-95-186. En entornos propensos a error, es necesario que el decodificador sea tolerante a trenes binarios recibidos con errores. En la práctica, suele haber mecanismos de la jerarquía de protocolo de transporte que proporcionan información acerca de las partes del tren binario dañadas o perdidas, tal como el número de secuencia RTP o la comprobación CRC AL3 de H.223. Puede obtenerse una calidad de funcionamiento razonable dividiendo las imágenes codificadas en segmentos decodificables (más o menos) independientemente, con cabeceras de GOB o rebanadas, incluso sin los mecanismos básicos de recuperación de errores del propio decodificador de vídeo.

En los supuestos de aplicación propensos a error descritos en III.2.3 y III.2.4, nunca se presentan errores en los bits del tren binario porque la pila de transporte ya garantiza que los entornos están libres de errores binarios gracias a las comprobaciones CRC. Por consiguiente, la única posibilidad de errores sintácticos es la pérdida de información entre marcas de sincronización (imagen y cabeceras de GOB o rebanada). Este error debe detectarse durante la decodificación, para aplicar la ocultación de errores descrita en III.5.4 a los macrobloques perdidos. Cuando se pierda una cabecera de imagen se sustituirá por la cabecera de la imagen anterior o por un duplicado de la cabecera de imagen cuando exista (por ejemplo en la cabecera de la parte útil del RTP).

Cuando haya errores en los bits del tren binario recibido, el decodificador de vídeo debe detectarlos como violaciones sintácticas o semánticas. A saber:

- Aparece una palabra código ilegal. Ésta es la causa más frecuente de suspensión de la decodificación.
- No hay marca de sincronización tras la reconstrucción de la línea macrobloque. (Si no se utiliza GBSC en todas las filas de macrobloques, esta condición no puede comprobarse.)

- Los vectores de movimiento apuntan más allá del intervalo admisible.
- La posición de los puntos de los coeficientes DCT reconstruidos está más allá de la posición 63, no estando en vigor el anexo S.
- Los valores de la crominancia en continua están fuera del intervalo normal.

Los dos primeros puntos son los más importantes. La lista de comprobaciones podría extenderse considerablemente. Cuando se detectan estos errores, el decodificador de vídeo busca el siguiente punto de sincronización del tren binario. Todos los datos entre marcas de sincronización donde se detectó el error se ignoran (es decir no se copian a la memoria de trama) y se ocultan con arreglo a lo descrito en III.5.4.

III.5.4 Ocultación de errores

El siguiente algoritmo se considera eficaz cuando rige el anexo J y se utiliza el posfiltrado adecuado, definido en III.5.1. El interfuncionamiento con el posfiltro del que trata III.5.2, queda en estudio.

Esta cláusula se basa, en parte, en el modelo de ocultación de errores TCON descrito en LBC-95-186. Los macrobloques perdidos se ocultan utilizando un algoritmo muy sencillo aunque eficaz. En las partes de la imagen afectadas por la pérdida de datos, se utilizan datos de la anterior imagen codificada para la ocultación. No se aplica la ocultación de errores cuando se pierde parte de la primera imagen recibida. Al vector de movimiento del macrobloque perdido se le da el valor del vector de movimiento del macrobloque que está encima del perdido. Cuando este vector de movimiento no está disponible, se pone el vector de movimiento a *ceró*. Con el vector de movimiento adecuado, el macrobloque correspondiente a la imagen anterior en la posición espacial definida por este vector de movimiento se copia a la posición del macrobloque perdido de la imagen actual.

Casos particulares correspondientes a 4MV y OBMC:

- Si el macrobloque que está encima del perdido se hubiera codificado con cuatro vectores de movimiento (como podría ocurrir cuando rigen los anexos F o J), la ocultación de errores funciona igual que cuando se utilizan los bloques de 8×8 , de modo que:
 - los dos bloques de 8×8 a la izquierda del macrobloque perdido utilizan el vector de movimiento del bloque de 8×8 a la izquierda del fondo del macrobloque superior, y análogamente;
 - los dos bloques de 8×8 a la derecha del macrobloque perdido utilizan el vector de movimiento del bloque de 8×8 a la derecha del fondo del macrobloque superior.
- Cuando rige el anexo F, se aplica la ponderación OBMC a los vectores de movimiento de ocultación.

III.6 Sección de captura de información

Esta cláusula contiene toda la información oficialmente aprobada para su incorporación al modelo de prueba por parte del Grupo de Expertos en codificación del UIT-T, pero que no encaja en los supuestos de aplicación definidos en III.2. La mayor parte de esta información sólo es aplicable a determinados modos ópticos, tales como el modo escalabilidad (anexo O), las tramas PB (anexos G y M) y el modo actualización de resolución reducida (anexo Q). No se ha comprobado totalmente la compatibilidad de toda la información aquí consignada con los algoritmos principales utilizados en los supuestos de aplicación. Se ha solicitado información complementaria.

III.6.1 El modo de tramas PB (anexos G y M)

El anexo M ofrece mejores opciones de predicción que el anexo G y debe utilizarse en vez del anexo G siempre que rija la versión 2 de H.263.

III.6.1.1 Estimación del movimiento de las tramas PB y decisión del modo mejorados

Los vectores de movimiento candidatos en sentido directo e inverso para cada uno de los bloques del macrobloque B se obtienen por ajuste del mejor vector de movimiento, MV, del macrobloque P, de acuerdo con lo indicado en H.263. Estos vectores se utilizan para ejecutar una predicción bidireccional en la búsqueda de la suma cuadrática de las diferencias bidireccionales, de acuerdo con lo descrito en el anexo M de H.263, modo tramas PB mejoradas, aunque con MVD igual a cero.

Después, para el macrobloque B de 16×16 se efectúa una estimación de movimiento normal de medio píxel y de píxel entero, relativa a la anterior imagen P reconstruida. La mejor suma cuadrática de las diferencias en sentido directo para esta estimación de movimiento se compara con la suma cuadrática de las diferencias bidireccionales de la predicción bidireccional. Si la suma cuadrática de las diferencias bidireccionales menos la suma cuadrática de las diferencias en sentido directo es mayor que 100 se escoge para este macrobloque la predicción en sentido directo. En tal caso, el vector de movimiento en sentido directo calculado a partir de la estimación de movimiento anterior, se transmite directamente en MVDB, sin predicción de vector de movimiento. Si resulta mejor la predicción bidireccional, no se transmite MVDB.

III.6.2 Modo escalabilidad (anexo O)

III.6.2.1 Estimación del movimiento de la trama B y decisión del modo verdaderos

Las tramas B admiten varias opciones de predicción, y por consiguiente varios modos de codificación. La estimación de movimiento directo se realiza respecto a la anterior trama I/P reconstruida, mientras que la estimación del movimiento inverso se efectúa respecto a la futura trama I/P reconstruida.

Por otra parte, las tramas B admiten la predicción de modo inmediato, que no requiere la transmisión de datos del vector de movimiento, y la predicción de modo bidireccional, que utiliza los vectores de movimiento resultantes de las estimaciones de movimiento directo e inverso. Para la predicción en modo inmediato, se ajusta el vector de movimiento existente entre el par de tramas I/P o el P/P.

Los modos de codificación más interesantes son, por este orden, el inmediato, el directo, el inverso, y el bidireccional. Para mostrar el grado de interés de cada uno de los modos de predicción posibles, se calcula la suma cuadrática de las diferencias (SAD) correspondiente, ajustándose a continuación. El modo inmediato se favorece restando 100 a su SAD. El modo directo se favorece restando 50 a su SAD. La SAD del modo en sentido inverso no se modifica. El modo bidireccional se penaliza añadiendo 75 a su SAD. El modo de menor SAD, una vez introducidas estas modificaciones, se toma como modo de codificación.

III.6.2.2 Estimación del movimiento de la trama EI/EP y decisión del modo

Las tramas EI y EP también admiten varias opciones de predicción, y por consiguiente varios modos de codificación. La estimación de movimiento directo se realiza respecto a la anterior trama EI/EP reconstruida.

Por otra parte, las tramas EI y EP admiten la predicción de modo ascendente, que no requiere la transmisión de datos del vector de movimiento, y la predicción de modo bidireccional, que utiliza los vectores de movimiento resultantes de la estimación de movimiento directa. La predicción en modo ascendente, se efectúa desde la misma posición espacial de la trama reconstruida, y acaso no muestreada, de la capa de referencia.

Los modos de codificación más interesantes son, por este orden, el ascendente, el directo y el bidireccional. Para mostrar el grado de interés de cada uno de los modos de predicción posibles, se calcula la suma cuadrática de las diferencias (SAD) correspondientes, ajustándose a continuación. El modo en sentido ascendente se favorece restando 50 a su SAD. La SAD del modo en sentido directo no se modifica. El modo bidireccional se penaliza añadiendo 100 a su SAD. El modo de menor SAD, una vez introducidas estas modificaciones, se toma como modo de codificación.

III.6.2.3 Control de velocidad de las tramas P y B

III.6.2.3.1 Nivel de macrobloque

El control de velocidad a nivel del macrobloque descrito en III.4.2.2 puede utilizarse directamente con las tramas B. La única diferencia es que al ser distintas las estadísticas de las tramas B y las de las tramas P, los parámetros de control de velocidad K y C (que se actualizan en cada macrobloque) toman sus valores en intervalos distintos. Por consiguiente, cuando se utilicen tramas P y B deberán utilizarse parámetros distintos $\{K_P, C_P\}$ y $\{K_B, C_B\}$ para las tramas P y B, respectivamente.

III.6.2.3.2 Nivel de trama

El control de velocidad a nivel de trama de III.4.2.1 establece para la trama P (tras la primera trama I) un objetivo de número de bits casi constante, lo que constituye una estrategia eficaz para las comunicaciones de vídeo de bajo retardo. Sin embargo, en los supuestos en los que se inserta una o varias tramas B entre las P, hay que utilizar una técnica que permita asignar menos bits a las tramas B, debido a que las tramas B son más fáciles de codificar.

Esta subcláusula describe una técnica idónea para la asignación de un objetivo de número de bits a las tramas P y B. La génesis de este método se describe en [Q15-C-19] [Q15-D-22]. Este método deriva del caso típico en el que el patrón de los tipos de trama es:

$$I, B, \dots, B, P, B, \dots, B, P, B, \dots, B, P, B, \dots, B, P, \dots$$

El conjunto de tramas "B, ..., B, P" se repite periódicamente tras la primera trama I y recibe el nombre de grupo de imágenes o GOP. Si M_B es el número de tramas B de un GOP, el objetivo del número de bits para la imagen P de dicho GOP, T_P , y el objetivo para cada una de las tramas B, T_B , puede calcularse del siguiente modo:

$$T_P = T - M_B T_B$$
$$T_B = \frac{T - 16^2 N(C_P - \beta C_B)}{\beta + M_B}$$
$$\beta = 0,9 \beta_{PREV} + 0,1F \frac{E_P}{E_C}$$

definiéndose los parámetros de las anteriores ecuaciones del siguiente modo:

- T , M , y N son, respectivamente, el número de bits del GOP, el número de tramas del GOP, y el número de macrobloques de una trama.
- El valor de β determina cuántos bits se asignan a la trama P y cuántos a las B. β aumenta con F y E_P/E_B , descritos a continuación.
- F determina el valor de la PSNR (relación señal/ruido de pico) de las tramas P en comparación con la de las tramas B. Por ejemplo, si F es igual a 1, la PSNR correspondiente a ambos tipos de trama será semejante y si F es mayor que 1, la PSNR de las P será superior que la de las B. El valor de F se calcula mediante la siguiente fórmula:

$$F = \max \left\{ \min \left(\frac{1,4}{\sqrt{Bpp}} - 0,3, 5 \right), 1 \right\}$$

siendo Bpp la velocidad de la secuencia de vídeo expresada en bits por píxel. Con esta relación, la PSNR de las tramas P es 1 dB superior a la de las tramas B, por término medio, lo que constituye un compromiso razonable.

- E_P es la energía de la trama P del anterior GOP, definiéndose la energía como la suma de las varianzas de los errores de predicción del macrobloque, es decir,

$$E_P = \sum_{i=1}^N \sigma_i^2$$

siendo σ_i^2 la varianza del i -ésimo macrobloque de la trama P (anterior), definida en III.4.2.2. Por otra parte, E_B es la media de las energías de las tramas B del GOP anterior, o sea,

$$E_B = \frac{1}{M_B} \sum_{m=1}^{M_B} E_{B,m}$$

siendo $E_{B,m}$ la energía de la m -ésima trama B del anterior GOP.

- Se otorga a β_{prev} el valor F en el primer GOP y el anterior valor de β en los GOP subsiguientes.
- C_P y C_B son las velocidades de movimiento y sintaxis (en bits por píxel) correspondientes a las tramas P y B, respectivamente, obteniéndose sus valores a partir del control de velocidad de los respectivos niveles de macrobloques descritos en III.4.2.2.

El anterior control de velocidad a nivel de trama, descrito en III.4.2.1, estaba destinado a los GOP del tipo "P...P", y corresponde al caso específico en el que $E_P = E_B$, $F = 1$, (o lo que resulta equivalente, $\beta = 1$) y $C_P = C_B$ en las anteriores ecuaciones.

Por último, antes de codificar una cierta trama (ya sea con un objetivo de T_P o T_B bits), se resta el valor Δ de la velocidad binaria objetivo, definida en III.4.2.1, que proporciona información del estado de saturación de la memoria intermedia del codificador y del umbral de saltos de trama. A este último se le ha dado el valor de la velocidad binaria del canal (en bits por segundo) dividida por la velocidad de codificación de tramas lo que constituye una elección acertada para los supuestos de bajo retardo. No obstante, cuando se insertan tramas B entre las P, y por lo tanto el retardo no es importante, podría resultar conveniente un umbral de saltos de trama ligeramente superior (y un mayor valor de la A definida en III.4.2.1).

III.6.2.3.3 SNR y control de velocidad de la capa de mejora espacial

La velocidad binaria de las capas de mejora suele estar determinada por la aplicación específica. En cada capa, el control de velocidad puede utilizarse de forma equivalente a la velocidad binaria de la capa base, su velocidad de tramas y el patrón de GOP de la capa en cuestión. La única diferencia es que, como las tramas presentan estadísticas distintas en las diversas capas, debe haber variables específicas para K y C en cada capa. Más concretamente, debe haber parámetros específicos K y C para las diferentes capas y, dentro de cada una de ellas, K y C específicas para cada tipo de trama.

III.6.2.4 Utilización en entornos propensos a error

Además de otros métodos de recuperación de errores descritos en este apéndice, los números de imagen definidos en el anexo U (Selección de imagen de referencia mejorada) y el anexo W (Especificación adicional de información de mejora suplementaria) proporcionan un método de control de errores particularmente idóneo para el anexo O. Los números de imagen constituyen una parte integral del tren binario cuando rige el anexo U. De lo contrario se recomienda utilizar los números de imagen definidos en el anexo W.

La definición de números de imagen de los anexos U y W puede enunciarse del siguiente modo: el número de imagen deberá incrementarse en una unidad para cada I, P, PB, EI e imágenes EP codificada y transmitida, respecto a la anterior imagen almacenada en la misma capa de mejora. En

las imágenes B, el número de imagen se incrementará respecto al valor propio de la imagen no B más recientemente almacenada en la capa de referencia de la imagen B que precede a la imagen B en orden de tren binario (imagen que es inmediatamente posterior en el tiempo a la imagen B). Cuando las imágenes adyacentes de la misma capa de mejora tengan la misma referencia temporal, estando vigente un modo de selección de imagen de referencia, el decodificador se comportará como si hubiere recibido una indicación del envío de duplicados con el mismo contenido de imagen, aproximadamente, compartiendo todos ellos el mismo número de imagen.

Cuando la diferencia entre los números de dos imágenes recibidas y almacenadas consecutivamente en la misma capa de mejora no sea la unidad, no representando las imágenes las mismas escenas, como se ha indicado anteriormente, el decodificador deberá asumir la pérdida de imágenes o la corrupción de datos. En este caso, puede enviarse al codificador del extremo lejano por el canal de respaldo el mensaje definido en el anexo U o indicación similar que notifique la pérdida de imágenes. Si las imágenes perdidas pertenecieran a las capas de mejora, el decodificador podría continuar procesando la capa de base y las capas de mejora inferiores a la corrupta. Asimismo, el decodificador podría seguir decodificando la capa que hubiera perdido imágenes y cualquier otra superior, especialmente si se señala una imagen de referencia de reserva conveniente (véase W.6.3.13/H.263).

III.6.3 Modo actualización de resolución reducida (anexo Q)

Esta cláusula describe cómo utilizar el modo de actualización de resolución reducida definido en el anexo Q. La información adjunta no ha sido verificada.

III.6.3.1 Estimación del movimiento y selección del modo

En el modo actualización de resolución reducida, la estimación del movimiento se realiza sobre el macrobloque de luminancia de 32×32 en vez de sobre el macrobloque de 16×16 . Como medida del error se utiliza la suma de diferencia absoluta (SAD, *sum of absolute difference*). En este modo cada componente del vector de movimiento del macrobloque o de los cuatro vectores de movimiento, se limita a medio entero o al valor cero a fin de ampliar el intervalo de búsqueda con el mismo cuadro MVD.

III.6.3.2 Estimación del movimiento en el modo básico (sin opciones)

III.6.3.2.1 Estimación del movimiento en píxeles enteros

La búsqueda se efectúa con el desplazamiento en píxeles enteros sobre la componente Y. Las comparaciones se efectúan entre el macrobloque de entrada y el macrobloque desplazado de la imagen reconstruida anterior. Se utiliza una búsqueda completa, siendo el área de búsqueda de ± 30 píxeles como máximo en sentido horizontal y vertical en torno a la posición inicial del macrobloque.

$$SAD(x,y) = \sum_{i=1, j=1}^{32,32} |original - decodificada_anterior|_{x,y} = \text{"máximo } \pm 30\text{"}$$

Para el vector cero, SAD(0,0) se reduce en **400** a fin de favorecerlo cuando no haya ninguna diferencia importante.

$$SAD(0,0) = SAD(0,0) - 400$$

El par (x, y) obtenido a partir de la SAD mínima se toma como vector de movimiento de píxel entero, MV0. La SAD correspondiente es SAD(x,y).

III.6.3.2.2 Decisión del modo INTRA/INTER

Tras la estimación del movimiento en píxeles enteros, el codificador toma la decisión de utilizar la predicción INTRA o la INTER para la codificación. Para adoptar la decisión INTRA/INTER se calcularán los siguientes parámetros:

$$MB_media = \frac{\sum_{i=1, j=1}^{32,32} original}{1024}$$
$$A = \sum_{i=1, j=1}^{32,32} |original - MB_media|$$

Se adoptará el modo INTRA cuando: $A < (SAD(x, y) - 2000)$.

Obsérvese que cuando se utiliza $SAD(0,0)$ su valor ya se ha reducido previamente en 400.

Si se adopta el modo INTRA, ya no se necesitan más operaciones para la búsqueda del movimiento. Si se selecciona el modo INTER continúa la búsqueda del movimiento con una búsqueda de medio píxel en torno a la posición MV0.

III.6.3.2.3 Búsqueda de medio píxel

La búsqueda de medio píxel se efectúa con la trama reconstruida anterior, sobre la componente Y del macrobloque. El área de búsqueda es ± 1 medio píxel en torno a la matriz objetivo de 32×32 a la que apunta MV0, cumpliendo la condición de que cada componente del vector candidato de la búsqueda de medio píxel tenga un valor de medio entero o cero. Para el vector cero (0,0), $SAD(0,0)$ se reduce en 400 para la búsqueda entera.

Los valores de medio píxel se calculan de acuerdo con lo descrito en 6.1.2/H.263.

El vector resultante de la mejor concordancia durante la búsqueda de medio píxel recibe el nombre de MV. MV tiene componente horizontal y vertical (MVx, MVy) que se miden en unidades de medio píxel.

III.6.3.3 Estimación de movimiento en el modo predicción avanzada (anexo F)

Esta cláusula sólo se utiliza cuando se ha seleccionado el modo de predicción avanzada.

III.6.3.3.1 Estimación de movimiento de píxel entero

La búsqueda de píxel entero ± 2 dentro de $[-31, 30]$ se efectúa para los **16 x 16 bloques** en torno al vector entero de 32×32 .

III.6.3.3.2 Búsqueda de medio píxel

La búsqueda de medio píxel se efectúa para cada uno de los bloques en torno al vector entero de 16×16 . El área de búsqueda es $\pm 0,5$ píxeles en torno al vector entero de 16×16 del bloque correspondiente, cumpliendo la condición de que cada componente del vector candidato para la búsqueda del medio píxel tiene un valor de medio entero o cero y está dentro de $[-31,5, 30,5]$.

III.6.3.3.3 Decisión entre uno y cuatro MV en el modo de predicción avanzada

Esta subcláusula sólo es válida cuando se ha seleccionado el modo de predicción avanzada.

SAD del el mejor vector **32 x 32 MB** de medio píxel (incluida la resta de **400** si el vector es (0,0)):

$$SAD_{32}(x, y)$$

SAD del macrobloque completo para los mejores vectores de **bloque de 16 × 16** de medio píxel:

$$SAD_{4 \times 16} = \sum_1^4 SAD_{16}(x, y)$$

se aplica la siguiente regla:

Si: $SAD_{4 \times 16} < SAD_{32} - 800$, se utilizará la predicción de **bloque de 16 × 16**

de lo contrario: se utilizará la predicción de **MB de 32 × 32**.

III.6.3.4 Estimación de movimiento en el modo vector de movimiento sin restricción (anexo D)

Esta cláusula sólo es aplicable si se selecciona el intervalo ampliado del vector de movimiento en el modo UMV.

III.6.3.4.1 Limitación de la ventana de búsqueda

Como la ventana que contiene los vectores de movimiento permitidos en este modo está centrada en torno al predictor del vector de movimiento en el macrobloque actual, se aplican ciertas restricciones a la búsqueda del vector de movimiento entero a fin de garantizar que los vectores de movimiento calculados serán transmisibles.

Con estas restricciones, tanto el vector **MB de 32 × 32** como los vectores de **bloque de 16 × 16** calculados mediante el procedimiento descrito a continuación, serán transmisibles, independientemente del predictor resultante del vector de movimiento de medio píxel de exactitud para el macrobloque o para cada uno de los cuatro bloques.

III.6.3.4.2 Búsqueda de píxel entero

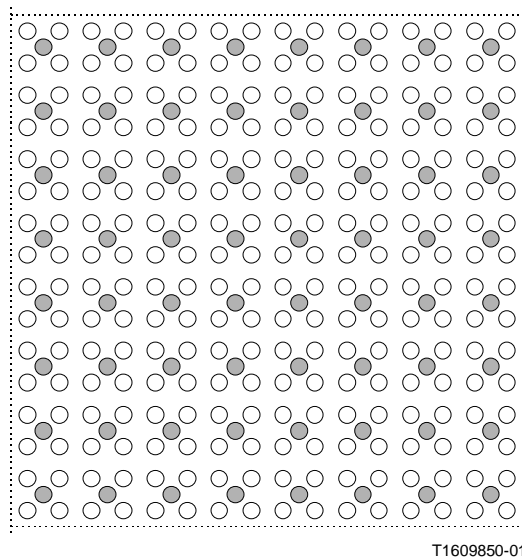
En primer lugar se calcula el predictor del vector de movimiento correspondiente al vector **MB 32 × 32** basado exclusivamente en vectores de movimiento enteros. A continuación la búsqueda **MB de 32 × 32** se centra en torno al predictor truncado, con una ventana de búsqueda algo limitada. En el caso de cuatro vectores, la ventana de búsqueda **MB de 32 × 32** se limita al intervalo $29 - (2 * \text{ventana_búsqueda_bloque_16} \times 16 + 1)$. Como en este modelo la ventana de búsqueda **bloque_16 × 16** es 2,5, la ventana de búsqueda por defecto del **MB de 32 × 32** en el modo UMV resulta ser ± 23 posiciones enteras. A continuación se centran las búsquedas de **bloque_16 × 16** en torno al mejor vector **MB 32 × 32** y se efectúa la búsqueda de ± 2 píxels en cada **bloque_16 × 16**.

III.6.3.4.3 Búsqueda de medio píxel

Las búsquedas de medio píxel se realizan como en los otros modos. El área de búsqueda es $\pm 0,5$ píxels en torno al mejor vector entero del correspondiente macrobloque/bloque, cumpliendo la condición de que cada componente del vector candidato para la búsqueda de medio píxel tiene un valor de medio entero o cero.

III.6.3.5 Muestreo reducido del error de predicción

Tras la compensación de movimiento en base a bloques de 16×16 , se circunscribe el muestreo del bloque de error de predicción de 16×16 a un bloque de error de predicción de resolución reducida de 8×8 . Para realizar esta sencilla implementación, se limita el filtrado a un bloque que permite el muestreo ampliado en base a un bloque individual. En la figura III.5 se puede ver la posición de las muestras. El procedimiento de muestreo reducido de los píxels de luminancia y crominancia se define en la figura III.6. El filtrado no tiene en cuenta los límites de los bloques. En la figura III.6 "/" indica la división con truncamiento.



- Posición de las muestras en el bloque de error de predicción de resolución reducida de 8×8
- Posición de las muestras en el bloque de error de predicción de 16×16
- Borde del bloque

Figura III.5/H.263 – Posición de las muestras en el bloque de error de predicción de resolución reducida de 8×8 y en el bloque de error de predicción de 16×16

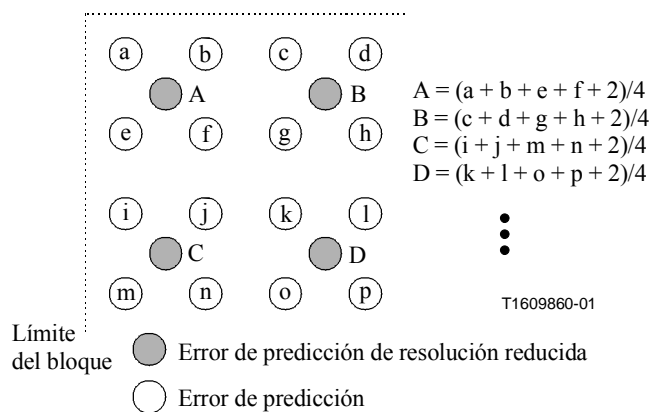


Figura III.6/H.263 – Creación de error de predicción de resolución reducida para los píxeles dentro de un bloque

III.6.3.6 Transformada y cuantificación

Se aplica una transformada de coseno discreto (DCT, *discrete cosine transform*) bidimensional separable al bloque de error de predicción de resolución reducida de 8×8 , al igual que en el modo por defecto. A continuación se efectúa la cuantificación como se describió en el modo por defecto.

III.6.3.7 Conmutación

Esta cláusula describe un sencillo algoritmo de conmutación correspondiente al anexo Q.

NOTA – Este algoritmo podría aplicarse al apartado "factor de 4" del anexo P con ligeras modificaciones.

III.6.3.7.1 Algoritmo de decisión de la resolución

Para decidir la resolución a la que alude el anexo Q, se utiliza un sencillo algoritmo de decisión basado en \overline{QP}_{i-1} y B_{i-1} .

\overline{QP}_{i-1} QP media de la trama codificada anterior;

B_{i-1} número de bits utilizados por la trama codificada anterior.

Suponiendo que la relación entre \overline{QP}_{i-1} y B_{i-1} se aproxima a la proporción inversa, el producto de \overline{QP}_{i-1} y B_{i-1} puede considerarse un índice de la complejidad aproximada de la trama codificada.

Si la trama actual se encuentra en el modo por defecto, se efectúa la conmutación al modo de actualización de resolución reducida cuando el producto de \overline{QP}_{i-1} y B_{i-1} sea mayor que el umbral TH1.

Si la trama actual está en el modo de actualización de resolución reducida, la conmutación al modo por defecto se realiza si el producto de \overline{QP}_{i-1} y B_{i-1} es menor que el umbral TH2.

Conmutación del **modo por defecto** al **modo actualización de resolución reducida**

```
si( $\overline{QP}_{i-1} * B_{i-1} > TH1$ ){  
    conmutar a  
    modo de actualización de resolución reducida;  
}
```

Conmutación del **modo de actualización de resolución reducida** al **modo por defecto**

```
si( $\overline{QP}_{i-1} * B_{i-1} < TH2$ ){  
    Conmutar a modo por defecto;  
}
```

TH1 se determina en la siguiente ecuación, en la que QP1 y FR1 representan la calidad subjetiva mínima admisible para la codificación en el modo por defecto.

$$TH1 = QP1 * (\text{Velocidad binaria_objetivo} / FR1)$$

Del mismo modo, se determina TH2 en la siguiente ecuación, representando QP2 y FR2 la calidad subjetiva máxima admisible para la codificación en el modo de actualización de resolución reducida.

$$TH2 = QP2 * (\text{Velocidad binaria_objetivo} / FR2)$$

Los valores de QP1, FR1, QP2, y FR2 pueden depender del formato fuente, la velocidad de la trama objetivo, y la velocidad binaria objetivo. Si el formato fuente indica CIF, la velocidad de trama objetivo es de 10 tramas por segundo, y la velocidad binaria objetivo de 48 kbit/s, por lo que QP1 = 16, FR1 = 7, QP2 = 7, FR2 = 9, respectivamente.

III.6.3.8 Restricción de los coeficientes DCT en la conmutación de resolución reducida a resolución normal

Una vez seleccionado el modo resolución reducida, es posible que se pierdan detalles de la imagen. Si el modo retorna al modo por defecto, hay que reproducir el detalle de la imagen consumiendo una gran cantidad de bits. El súbito incremento del número de bits de codificación suele provocar saltos de trama inesperados. Además, como el algoritmo de decisión de resolución descrito anteriormente

utiliza el producto de la QP media por el número de bits, este brusco aumento del número de bits provoca de nuevo la conmutación al modo de actualización de resolución reducida, produciéndose a menudo la oscilación entre ambos modos. Para evitar esta degradación, se introduce la restricción de los coeficientes DCT en varias tramas posteriores a la conmutación del modo de actualización resolución reducida al modo por defecto. En la primera trama posterior a la conmutación al modo por defecto, pueden enviarse solamente los coeficientes dentro de la frecuencia baja de 4×4 , a continuación de modo similar 5×5 para el segundo, 6×6 para el tercero y 7×7 para los sucesivos. Este algoritmo de "aterizaje suave" puede suprimir eficazmente el salto de tramas y la oscilación entre modos.

III.6.3.8.1 Control de velocidad

El control de velocidad es idéntico al del modo por defecto, excepto en el número de cuartos de macrobloques.

III.6.4 Búsqueda rápida mediante desigualdades matemáticas

Hay varios métodos conocidos que aceleran la búsqueda de movimiento, y están basados en desigualdades matemáticas [LS95] [LT97]. Estas desigualdades, por ejemplo la desigualdad triangular, proporcionan un límite inferior a la norma de diferencia entre vectores. En la concordancia entre bloques, el criterio de determinación de la distorsión que se suele utilizar es la suma de diferencias absolutas (SAD) y la suma de diferencias cuadráticas (SSD) entre la predicción compensada en movimiento $c[x, y]$ y la señal original $s[x, y]$. La incorporación de la desigualdad triangular a las sumas SAD y SSD da como resultado:

$$D(s, c) = \sum_{[x, y] \in B} |s[x, y] - c[x, y]|^p \geq \hat{D}(s, c) = \left| \left(\sum_{[x, y] \in B} |s[x, y]|^p \right)^{1/p} - \left(\sum_{[x, y] \in B} |c[x, y]|^p \right)^{1/p} \right|^p \quad (\text{III-6.4.1})$$

siendo $p = 1$ para SAD y $p = 2$ para SSD. Obsérvese que para $p = 2$, la desigualdad utilizada en [LT97] es diferente de la ecuación (III-6.4.1). En la práctica, se han observado pocas diferencias entre ambas desigualdades. En algunos bloques, la desigualdad utilizada por [LT97] proporciona un límite más exacto mientras que para otros bloques la desigualdad triangular se comporta mejor. El conjunto B comprende las posiciones de muestreo de los bloques considerados, por ejemplo un bloque de 16×16 muestras.

Si se supone que es D_{min} el valor de la distorsión mínima calculada anteriormente en la búsqueda de movimiento de bloque, se garantiza que la distorsión $D(s, c)$ de otro bloque c en nuestro intervalo de búsqueda excederá D_{min} si el límite inferior de $D(s, c)$ supera D_{min} . Para ser más precisos, hay que rechazar el bloque c cuando

$$\hat{D}(s, c) \geq D_{min} \quad (\text{III-6.4.2})$$

La estructura peculiar del problema de estimación de movimiento permite utilizar un método rápido de cálculo de los valores normales de todos los bloques $c[x, y]$ de las tramas decodificadas anteriores [LS95]. La ampliación a un criterio de estimación de movimiento de velocidad restringida es inmediata [CM97].

III.6.4.1 Orden de búsqueda

Es evidente que si se establece un valor pequeño para D_{min} al principio de la búsqueda se provoca un rechazo de muchos otros bloques más adelante reduciendo por consiguiente el cálculo. Por lo tanto, el orden de comprobación de los bloques en el intervalo de búsqueda tiene una gran repercusión en

el tiempo de cálculo. Por ejemplo, si se toman las tablas de códigos de Huffman para los vectores de movimiento como información previa sobre nuestro espacio de búsqueda, el orden de búsqueda de los vectores de movimiento debería ser el de velocidad binaria creciente. Esto incrementa la probabilidad de encontrar una buena concordancia al principio del proceso de búsqueda. Una buena aproximación de estas probabilidades es la búsqueda en espiral descrita en III.3.1.3.1.

III.6.4.2 Desigualdades triangulares múltiples

De acuerdo con [WZG99], pueden utilizarse desigualdades triangulares múltiples. Supongamos que B se divide en subconjuntos B_n de modo que

$$B = \bigcup_n B_n \text{ y } \bigcap_n B_n = \phi \quad (\text{III-6.4.3})$$

La desigualdad triangular (III-6.4.1) es válida para todos los subconjuntos posibles B_n . Es posible volver a escribir la fórmula de $D(s, c)$ del siguiente modo:

$$\sum_{[x,y] \in B} |s[x, y] - c[x, y]|^p = \sum_n \sum_{[x,y] \in B_n} |s[x, y] - c[x, y]|^p \quad (\text{III-6.4.4})$$

y la aplicación de la desigualdad triangular a todos los B_n da como resultado:

$$D(s, c) = \sum_{[x,y] \in B} |s[x, y] - c[x, y]|^p \geq \sum_n \left(\left(\sum_{[x,y] \in B_n} |s[x, y]|^p \right)^{1/p} - \left(\sum_{[x,y] \in B_n} |c[x, y]|^p \right)^{1/p} \right)^p \quad (\text{III-6.4.5})$$

Obsérvese que la ecuación (III-6.4.5) constituye un límite inferior más estricto que el de la ecuación (III-6.4.1) aunque requiere más cálculo. Por consiguiente, tiene en cuenta el compromiso entre definición del límite inferior y complejidad de cálculo.

Una cuestión importante a este respecto sigue siendo la selección de las particiones B_n . Si bien es cierto que la fórmula (III-6.4.5) da buenos resultados con todos los subconjuntos posibles que satisfacen la fórmula (III-6.4.4), como los valores normales de todos bloques del espacio de búsqueda han de ser precalculados puede aplicarse el método rápido descrito en [LS95]. Por consiguiente, la subdivisión aleatoria de B en n conjuntos arbitrarios puede ser una elección equivocada. Por contra, a efectos de cálculos, puede resultar más conveniente la subdivisión simétrica de B . En [LT97], se plantea la división de un bloque cuadrado de 16×16 en dos particiones diferentes. En la primera partición se crean 16 subconjuntos B_n en la que cada uno de ellos tiene 16 líneas con 16 muestras. En la segunda partición hay 16 subconjuntos B_n cada uno de ellos con 16 columnas que contienen 16 muestras.

Dado que H.263 admite los bloques de 8×8 , se aplica la solución propuesta en [LC95], en la que un bloque de 16×16 se descompone en subbloques. El bloque de 16×16 se divide en un conjunto de 16×16 muestras y cuatro subconjuntos de 8×8 muestras. Las distintas desigualdades triangulares (para cada subconjunto) se aplican en orden del tiempo de cálculo necesario para evaluarlas, es decir, en primer lugar se comprueba la desigualdad triangular de 16×16 y a continuación las desigualdades relativas a los bloques de tamaño 8×8 . En el nivel de bloque de 8×8 sólo se comprueba la desigualdad triangular de 8×8 .

III.6.5 Control de velocidad en la trama de codificación

Esta cláusula contiene dos algoritmos que pueden utilizarse en vez del algoritmo de control de velocidad de la capa de trama definido en III.4.2.1. Aunque el algoritmo de velocidad de trama fija es útil exclusivamente en los entornos donde una velocidad de trama completamente fija es más importante que la calidad y el retardo percibidos, el algoritmo de control adaptable incrementa la

calidad percibida aunque mantiene unas características de retardo razonables, ajustando la velocidad de la trama de acuerdo con el contenido de las escenas.

III.6.5.1 Control adaptable de velocidad de las tramas de codificación

El objetivo de este algoritmo de control de velocidad de la trama de codificación es mantener la calidad de las tramas P dentro del intervalo tolerable en condiciones de cambio brusco de movimiento y en entornos de canales de comunicaciones variables en el tiempo sin degradación manifiesta de la suavidad de movimiento percibida. El algoritmo adapta a la velocidad de la trama de codificación en base a la información de movimiento del vídeo subyacente para mantener la calidad de imagen de cada trama P dentro de un intervalo aceptable. Como resulta difícil obtener buena calidad tanto de resolución espacial como temporal (en términos de suavidad de movimiento) a muy bajas velocidades binarias, se adopta un control de velocidad de la trama de codificación buscando un compromiso entre la calidad espacial/temporal basado del movimiento del vídeo y el ancho de banda disponible en el canal. Se ha observado que el ojo humano es muy sensible a los cambios bruscos de velocidad de la trama de codificación (o del intervalo de velocidades correspondiente). Este esquema pretende reducir la degradación temporal en términos de las alteraciones del movimiento percibidas por el ojo humano. Además, se impone un retardo temporal de codificación lenta al procesamiento en tiempo real. El control de velocidad de la trama de codificación en CBR con variación temporal y la relación con los llamados algoritmos de control de velocidad TMN8 de III.4.2 es objeto de explicación en los párrafos siguientes. El desarrollo de este algoritmo se describe en [Q15-G-22].

Se utilizan las siguientes definiciones:

i : Índice de la trama codificada.

a, b, a' y b' : Coeficientes del modelo R-D de la capa de trama.

f_{i-1} : Trama de referencia reconstruida en el instante temporal previo.

f_i : Imagen descomprimida en el instante de tiempo actual.

\overline{QP}_i : QP promedio de todos los macrobloques de una trama.

$\hat{B}(Q), \hat{D}(Q)$: Modelos de velocidad y distorsión de una trama, respectivamente.

$MAD(f_{i-1}, f_i)$: Media de la diferencia absoluta entre f_{i-1} y f_i .

R_i : Ancho de banda actual disponible en el canal.

F_i : Intervalo de trama de codificación actual con la hipótesis de que la cámara captura las tramas a la velocidad de G tramas por segundo.

Paso 0: Fijar los parámetros iniciales tales como la velocidad de trama de codificación inicial F_0 , el umbral de ajuste del intervalo de trama de codificación $c = 0,04$. Además, los tamaños de la tabla R-D de la capa de trama T_0, T_{max} se ponen a 0 y 20, respectivamente. Por último la iteración empieza con $i = 1$.

Paso 1: Desplazar la tabla de la base de datos R-D haciendo $B_k = B_{k+1}, D_k = D_{k+1}, \overline{QP}_k = \overline{QP}_{k+1}$ para $k = 1, \dots, T_{i-1} - 1$.

A continuación, añadir un nuevo elemento al final de la tabla haciendo $B_{T_{i-1}} = B_{i-1}, D_{T_{i-1}} = D_{i-1}, \overline{QP}_{T_{i-1}} = \overline{QP}_{i-1}$.

Paso 2: Calcular los modelos de velocidad y distorsión respecto a la QP media de las tramas. En primer lugar calcular los coeficientes del modelo de velocidad y distorsión utilizando la anterior tabla R-D.

$$\begin{aligned}\hat{B}(QP_i) &= (a \cdot QP_i^{-1} + b \cdot QP_i^{-2}) \cdot MAD(f_{i-1}, f_i), \\ \hat{D}(QP_i) &= a' QP_i + b'.\end{aligned}\tag{III-6.5.1}$$

$$a = \frac{\sum_{k=1}^{T_{i-1}} \{R_k QP_k^{-1} - b QP_k^{-3}\}}{\sum_{k=1}^{T_{i-1}} QP_k^{-2}}, b = \frac{\left(\sum_{k=1}^{T_{i-1}} R_k QP_k^{-2}\right) \left(\sum_{k=1}^{T_{i-1}} QP_k^{-2}\right) - \left(\sum_{k=1}^{T_{i-1}} R_k QP_k^{-1}\right) \left(\sum_{k=1}^{T_{i-1}} QP_k^{-3}\right)}{\left(\sum_{k=1}^{T_{i-1}} QP_k^{-4}\right) \left(\sum_{k=1}^{T_{i-1}} QP_k^{-2}\right) - \left(\sum_{k=1}^{T_{i-1}} QP_k^{-3}\right)^2},\tag{III-6.5.2}$$

$$a' = \frac{\sum_{k=1}^{T_{i-1}} d_k \sum_{k=1}^{T_{i-1}} QP_k - T_{i-1} \sum_{k=1}^{T_{i-1}} d_k QP_k}{\left(\sum_{k=1}^{T_{i-1}} QP_k\right)^2 - T_{i-1} \sum_{k=1}^{T_{i-1}} QP_k^2}, b' = \frac{\sum_{k=1}^{T_{i-1}} d_k - a' \sum_{k=1}^{T_{i-1}} QP_k}{T_{i-1}},\tag{III-6.5.3}$$

A continuación, suprimir los valores más dispares utilizando la siguiente comprobación para $k = 1 : T_{i-1}$. Si $|\hat{B} - B_k| > \sigma_B$ o $|\hat{D} - D_k| > \sigma_D$ desactivar este dato temporalmente y repetir el paso 2 con los datos afinados.

Paso 3: Calcular la distorsión estimada mediante

$$\begin{aligned}\hat{D} &= a' \frac{a \cdot MAD(f_{i-1}, f_i) + \sqrt{(a \cdot MAD(f_{i-1}, f_i))^2 + 4b \cdot B(F_{i-1}) \cdot MAD(f_{i-1}, f_i)}}{2B(F_{i-1})} + b', \\ B(F_{i-1}) &= \frac{F_{i-1}}{G} \cdot R_i.\end{aligned}\tag{III-6.5.4}$$

Paso 4: Determinar el intervalo de trama de codificación calculando en primer lugar los valores umbrales mediante las fórmulas $TH_{d1} = (1+c) \cdot D_{avg}$, $TH_{d2} = (1-c/2) \cdot D_{avg}$, siendo D_{avg} la distorsión media de las cinco tramas codificadas anteriores. Obsérvese que cuando c es grande, el control de la velocidad de la trama de codificación queda desactivado y sólo funciona el control de velocidad de la capa de macrobloque de III.4.2.2. A continuación ajustar el intervalo de trama de codificación como se especifica en la ecuación (III-6.5.5) con $\Delta F_{i-1} = [0.3 \cdot F_{i-1}]$.

Paso 5: Llamar al control de velocidad de nivel de macrobloque de III.4.2.2 con la velocidad binaria objetivo \tilde{B}_i , que proporcionará \overline{QP}_i, B_i y D_i .

$$\tilde{B}_i = \begin{cases} (F_{i-1} + \Delta F_{i-1}) \cdot R_i / G & \text{if } \hat{D} > TH_{d1}, \\ (F_{i-1} + \Delta F_{i-1}) \cdot R_i / G & \text{if } \hat{D} < TH_{d2}, \\ F_{i-1} \cdot R_i / G & \text{en los demás casos} \end{cases}\tag{III-6.5.5}$$

Paso 6: Si se han codificado todas las tramas, parar. De lo contrario ir al paso 1 con $T_i = \min\{T_{i-1} + 1, T_{max}\}$ e $i = i + 1$.

III.6.6 Observaciones sobre el uso optimizado del modo selección de imagen de referencia mejorada

III.6.6.1 Técnicas de búsqueda rápida

Como la estimación de movimiento se amplía de una imagen a varias, se recomienda la utilización de técnicas de búsqueda rápida. Sírvanse consultar el documento [Q15-D-55] en donde se detallan las posibles reducciones en tiempo de cálculo.

III.6.6.2 Codificación de la recuperación de error

Al igual que para el MCP de imagen única, las pérdidas del contenido de imagen y su ocultación pueden provocar imágenes de referencia distintas en el codificador y el decodificador para MCP multi-imagen provocando la propagación de errores temporales. En ambos sistemas, la codificación de macrobloques en el modo INTRA puede detener la propagación temporal de errores. No obstante, en el MCP multi-imagen, la selección del vector del movimiento y el parámetro de referencia de imagen pueden tener asimismo una repercusión importante sobre la calidad de funcionamiento. Sírvanse consultar el documento Q15-H-24 en el que se detalla una estrategia de codificación posible.

III.7 Referencias

- [CM97] COBAN (M.), MERSEREAU (R.M.): Computationally Efficient Exhaustive Search Algorithm for Rate-Constrained Motion Estimation, in *Proc. ICIP*, Santa Barbara, Estados Unidos, octubre de 1997.
- [GCK99] GALLANT (M.), CÔTÉ (G.), KOSENTINI (F.): A Computation Constrained Block-based Motion Estimation Algorithm for Low Bit-rate Video Coding, *apto IEEE Transactions on Image Processing*, diciembre de 1999.
- [GFS97] GIROD (B.), FÄRBER (N.), STEINBACH (E.): Performance of the H.263 Video Compression Standard, *Journal of VLSI Signal Processing: Systems for Signal, Image, and Video Technology. Special Issue on Recent Development in Video: Algorithms, Implementation and Applications*. Vol. 17, No. 2/3, pp. 101-111, noviembre de 1997.
- [LBC-95-186] Telenor Research: Definition of an Error Concealment Model (TCON), *Contribution LBC-95-186 to meeting of ITU-T SGXV Experts Group for Very Low Bit Rate Visual Telephony (Periodo de estudios 1993-1996)*, Boston, junio de 1995.
- [LC95] LEE (C.-H.), CHEN (L.-H.): A Fast Search Algorithm for Vector Quantization Using Mean Pyramids of Codewords, en *IEEE TR-COM*, Vol. 43, No. 2/3/4, pp. 604-612, febrero/marzo/abril de 1995.
- [LS95] LI (W.), SALARI (E.): Successive Elimination Algorithm for Motion Estimation, *IEEE Trans. Image Proc.*, pp.105-107, enero de 1995.
- [LT97] LIN (Y.-C.), TAI (S.-C.): Fast Full-Search Block-Matching Algorithm for Motion-compensated Video Compression, en *IEEE TR-COM*, Vol. 45, No. 5, pp. 527-531, mayo de 1997.
- [Q15-A-20] RIBAS-CORBERA (J.), LEI (S.): Rate Control for Low-Delay Video Communications, *Contribution Q15-A-20 to ITU-T Video Coding Experts Group Meeting (UIT-T SG16 Q.15 Periodo de estudios 1997-2000)*, Portland, Oregon, Estados Unidos, junio de 1997.
- [Q15-B-23] CÔTÉ (G.), GALLANT (M.), KOSENTINI (F.): Experimental Results for Integer Pixel Fast Search Motion Estimation of TMN 8, *Contribution Q15-B-23 to ITU-T*

Video Coding Experts Group Second Meeting (UIT-T SG16 Q.15 Periodo de estudios 1997-2000), Sun River, Oregon, Estados Unidos, septiembre de 1997.

- [Q15-C-19] RIBAS-CORBERA (J.), LEI (S.): Extension of TMN8 rate control to B frames and enhancement layer, *Contribution Q15-C-19 to ITU-T Video Coding Experts Group Third Meeting (UIT-T SG16 Q.15 Periodo de estudios 1997-2000)*, Eibsee, Alemania, diciembre de 1997.
- [Q15-D-13] WIEGAND (T.), ANDREWS (B.): An Improved H.263 Coder Using Rate-Distortion Optimization, *Contribution Q15-D-13 to ITU-T Video Coding Experts Group Fourth Meeting (UIT-T SG16 Q.15 Periodo de estudios 1997-2000)*, Tampere, Finlandia, abril de 1998.
- [Q15-D-22] RIBAS-CORBERA (J.), LEI (S.): An improvement on the extension of TMN8 rate control to B frames, *Contribution Q15-D-22 to ITU-T Video Coding Experts Group Fourth Meeting (UIT-T SG16 Q.15 Periodo de estudios 1997-2000)*, Tampere, Finlandia, abril de 1998.
- [Q15-D-55] WIEGAND (T.), GIROD (B.), LINCOLN (B.), ANDREWS (B.): Fast Search for Long-Term Memory Motion-Compensated Prediction, *Contribution Q15-D-55 to ITU-T Video Coding Experts Group Fourth Meeting (UIT-T SG16 Q.15 Periodo de estudios 1997-2000)*, Tampere, Finlandia, abril de 1998.
- [Q15-D-58] CÔTÉ (G.), EROL (B.), GALLANT (M.), KOSENTINI (F.): H.263+: Video Coding at Low Bit Rates, *Contribution Q15-D-58 to ITU-T Video Coding Experts Group Fourth Meeting (UIT-T SG16 Q.15 Periodo de estudios 1997-2000)*, Tampere, Finlandia, abril de 1998.
- [Q15-E-15] WENGER (S.), CÔTÉ (G): Intra-Macroblock Refresh in Packet (Picture) Lossy Scenarios, *Contribution Q15-E-15 to ITU-T Video Coding Experts Group Fifth Meeting (UIT-T SG16 Q.15 Periodo de estudios 1997-2000)*, Whistler, British Columbia, Canadá, julio de 1998.
- [Q15-E-37] CÔTÉ (G.), WENGER (S.): Effects of standard-compliant macroblock intra refresh on rate-distortion performance, *Contribution Q15-E-37 to ITU-T Video Coding Experts Group Fifth Meeting (UIT-T SG16 Q.15 Periodo de estudios 1997-2000)*, Whistler, British Columbia, Canadá, julio de 1998.
- [Q15-G-22] SONG (H.), KIM (J.), KUO (C.-C.J.): Performance analysis of real-time encoding frame rate control proposal, *Contribution Q15-G-22 to ITU-T Video Coding Experts Group Seventh Meeting (UIT-T SG16 Q.15 Periodo de estudios 1997-2000)*, Monterey, California, Estados Unidos, febrero de 1999.
- [Q15-H-24] WIEGAND (T.), FARBER (N.), GIROD (B.): Error-Resilient Video Transmission Using Long-Term Memory Motion Compensation, *Contribution Q15-H-24 to ITU-T Video Coding Experts Group Eighth Meeting (UIT-T SG16 Q.15 Periodo de estudios 1997-2000)*, Berlin, Alemania, agosto de 1999.
- [Q15-I-26] HANNUKSELA (M.), LEDISCHKE (M.), ZHANG (J.): Results from Error Resilient Header Repetition Core Experiment, *Contribution Q15-I-26 to ITU-T Video Coding Experts Group Ninth Meeting (UIT-T SG16 Q.15 Periodo de estudios 1997-2000)*, Red Bank, New Jersey, Estados Unidos, octubre de 1999.
- [Q15-I-60] WENGER (S.): Common Conditions for Video Performance Evaluation in H.324/M error-prone systems, *Contribution Q15-I-60 to ITU-T Video Coding Experts Group Ninth Meeting (UIT-T SG16 Q.15 Periodo de estudios 1997-2000)*, Red Bank, New Jersey, Estados Unidos, octubre de 1999.

- [Q15-J-62] HANNUKSELA (M.): Picture header recovery for H.263 Test Model for H.324/M Use, *Contribution Q15-J-62 to ITU-T Video Coding Experts Group Tenth Meeting (UIT-T SG16 Q.15 Periodo de estudios 1997-2000)*, Osaka, Japón, mayo de 2000.
- [RFC 1889] SCHULZRINNE (H.), CASNER (S.), FREDERICK (R.), JACOBSON (V.): RTP: A transport Protocol for Real-Time Applications, *IETF RFC 1889*, enero de 1996.
- [RFC 2429] BORMANN (C.), CLINE (L.), DEISHER (G.), GARDOS (T.), MACIOCCO (C.), NEWELL (D.), OTT (J.), SULLIVAN (G.), WENGER (S.), ZHU (C.): RTP Payload Format for the 1998 Version of ITU-T Rec. H.263 Video (H.263+), *IETF RFC 2429*, octubre de 1998.
- [SG88] SHOHAM (Y.), GERSHO (A.): Efficient Bit Allocation for an Arbitrary Set of Quantizers, en *IEEE Trans. On Acoustics, Speech, and Signal Processing*, Vol. 36, No. 9, pp. 1445-453, septiembre de 1988.
- [WZG99] WIEGAND (T.), ZHANG (X.), GIROD (B.): Long-Term Memory Motion-Compensated Prediction, en *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 9, No. 1, pp. 70-84, febrero de 1999.

SERIES DE RECOMENDACIONES DEL UIT-T

Serie A	Organización del trabajo del UIT-T
Serie B	Medios de expresión: definiciones, símbolos, clasificación
Serie C	Estadísticas generales de telecomunicaciones
Serie D	Principios generales de tarificación
Serie E	Explotación general de la red, servicio telefónico, explotación del servicio y factores humanos
Serie F	Servicios de telecomunicación no telefónicos
Serie G	Sistemas y medios de transmisión, sistemas y redes digitales
Serie H	Sistemas audiovisuales y multimedia
Serie I	Red digital de servicios integrados
Serie J	Redes de cable y transmisión de programas radiofónicos y televisivos, y de otras señales multimedia
Serie K	Protección contra las interferencias
Serie L	Construcción, instalación y protección de los cables y otros elementos de planta exterior
Serie M	RGT y mantenimiento de redes: sistemas de transmisión, circuitos telefónicos, telegrafía, facsímil y circuitos arrendados internacionales
Serie N	Mantenimiento: circuitos internacionales para transmisiones radiofónicas y de televisión
Serie O	Especificaciones de los aparatos de medida
Serie P	Calidad de transmisión telefónica, instalaciones telefónicas y redes locales
Serie Q	Conmutación y señalización
Serie R	Transmisión telegráfica
Serie S	Equipos terminales para servicios de telegrafía
Serie T	Terminales para servicios de telemática
Serie U	Conmutación telegráfica
Serie V	Comunicación de datos por la red telefónica
Serie X	Redes de datos y comunicación entre sistemas abiertos
Serie Y	Infraestructura mundial de la información y aspectos del protocolo Internet
Serie Z	Lenguajes y aspectos generales de soporte lógico para sistemas de telecomunicación