

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **3 009 731**

51 Int. Cl.:

H04N 19/31 (2014.01)

H04N 19/70 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **15.06.2021** **E 23210896 (9)**

97 Fecha y número de publicación de la concesión europea: **15.01.2025** **EP 4300974**

54 Título: **Señalización de información del intervalo de obturador**

30 Prioridad:

15.06.2020 US 202016901911
25.03.2021 US 202117212701

45 Fecha de publicación y mención en BOPI de la
traducción de la patente:
31.03.2025

73 Titular/es:

DOLBY LABORATORIES LICENSING
CORPORATION (100.00%)
1275 Market Street
San Francisco, CA 94103, US

72 Inventor/es:

ATKINS, ROBIN;
YIN, PENG;
LU, TAORAN;
PU, FANGJUN;
MCCARTHY, SEAN THOMAS;
HUSAK, WALTER J.;
CHEN, TAO y
SU, GUAN-MING

74 Agente/Representante:

ELZABURU, S.L.P

ES 3 009 731 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

DESCRIPCIÓN

Señalización de información del intervalo de obturador

Tecnología

5 El presente documento se refiere, en general, a las imágenes. Más particularmente, una realización de la presente invención se refiere a la mensajería de información del intervalo de obturador.

Antecedentes

10 Como se usa en el presente documento, el término "rango dinámico" (Dynamic Range, DR) puede referirse a una capacidad del sistema visual humano (Human Visual System, HVS) para percibir un rango de intensidad (por ejemplo, luminancia, luma) en una imagen, por ejemplo, desde grises más oscuros (negros) a blancos más brillantes (resaltados). En este sentido, DR se refiere a una intensidad "relacionada con la escena". DR también puede referirse a la capacidad de un dispositivo de visualización para representar adecuada o aproximadamente un rango de intensidad de una amplitud particular. En este sentido, DR se refiere a una intensidad "referida a la pantalla". A menos que se especifique explícitamente que un sentido particular tiene una significación particular en cualquier punto de la descripción en el presente documento, debe inferirse que el término puede usarse en cualquier sentido, por ejemplo, indistintamente.

15 Como se usa en el presente documento, el término alto rango dinámico (High Dynamic Range, HDR) se refiere a una amplitud de DR que abarca los 14-15 órdenes de magnitud del sistema visual humano (HVS). En la práctica, el DR sobre el que un ser humano puede percibir simultáneamente una extensa amplitud en el rango de intensidad puede truncarse ligeramente, en relación con el HDR.

20 En la práctica, las imágenes comprenden uno o más componentes de color (por ejemplo, luma Y y croma Cb y Cr), en donde cada componente de color está representado por una precisión de n bits por cada píxel (por ejemplo, n = 8). Usando codificación de luminancia lineal, las imágenes en las que $n \leq 8$ (por ejemplo, imágenes JPEG de 24 bits de color) se consideran imágenes de rango dinámico estándar (Standard Dynamic Range, SDR), mientras que las imágenes en las que $n > 8$ pueden considerarse imágenes de rango dinámico mejorado. Las imágenes de HDR también pueden almacenarse y distribuirse usando formatos de punto flotante de alta precisión (por ejemplo, 16 bits), tales como el formato de archivo OpenEXR desarrollado por Industrial Light and Magic.

30 Actualmente, la distribución de contenido de vídeo de alto rango dinámico, tal como Dolby Vision de laboratorios Dolby, o HDR10 en Blue-Ray, se limita a resolución 4K (por ejemplo, 4096 x 2160 o 3840 x 2160, y similares) y 60 fotogramas por segundo (fps) por las capacidades de muchos dispositivos de reproducción. En versiones futuras, se anticipa que puede estar disponible contenido de hasta 8K de resolución (por ejemplo, 7680 x 4320) y 120 fps para distribución y reproducción. Es deseable que los tipos de contenido futuros sean compatibles con los dispositivos de reproducción existentes con el fin de simplificar un ecosistema de contenido de reproducción de HDR, tal como Dolby Vision. Idealmente, los productores de contenido deberían poder adoptar y distribuir futuras tecnologías de HDR sin tener que deducir y distribuir también versiones especiales del contenido que sean compatibles con los dispositivos de HDR existentes (tales como HDR10 o Dolby Vision). Como se aprecia por parte de los inventores en este caso, se desean técnicas mejoradas para la distribución escalable de contenido de vídeo, especialmente contenido de HDR.

35 Los enfoques descritos en esta sección son enfoques que podrían perseguirse, pero no necesariamente enfoques que se han concebido o perseguido anteriormente. Por lo tanto, a menos que se indique lo contrario, no debe asumirse que ninguno de los enfoques descritos en esta sección se califica como técnica anterior simplemente en virtud de su inclusión en esta sección. De manera similar, los problemas identificados con respecto a uno o más enfoques no deben suponer que se hayan reconocido en ninguna técnica anterior basándose en esta sección, a menos que se indique lo contrario.

40 Mccarthy et al: "Shutter interval information SEI message for HEVC (Draft 2)", Reunión del MPEG 129, es un texto del borrador que contiene cambios en el estándar Codificación de Vídeo de Alta Eficiencia (High Efficiency Video Coding, HEVC) para especificar el mensaje de SEI de información de intervalo del obturador.

Breve descripción de los dibujos

Una realización de la presente invención se ilustra a modo de ejemplo, y no a modo de limitación, en las figuras de los dibujos adjuntos, y en los que números de referencia similares se refieren a elementos similares, y en los que:

la figura 1 representa un proceso de ejemplo para un conducto de entrega de vídeo;

50 la figura 2 representa un proceso de ejemplo de combinación de fotogramas originales consecutivos para renderizar una velocidad de fotograma objetivo en un ángulo de obturador objetivo;

la figura 3 muestra una representación de ejemplo de una secuencia de entrada con velocidad de fotograma de entrada variable y ángulo de obturador variable en un contenedor con velocidad de fotograma fija; y

la figura 4 representa una representación de ejemplo para la escalabilidad temporal a diversas velocidades de fotograma y ángulos de obturador con compatibilidad con versiones anteriores.

Descripción de realizaciones de ejemplo

En el presente documento se describen realizaciones de ejemplo que se refieren a la escalabilidad de velocidad de fotograma para la codificación de vídeo. En la siguiente descripción, con fines explicativos, se exponen numerosos detalles específicos con el fin de proporcionar una comprensión exhaustiva de las diversas realizaciones de la presente invención. Será evidente, sin embargo, que las diversas realizaciones de la presente invención pueden ponerse en práctica sin estos detalles específicos. En otros casos, las estructuras y dispositivos bien conocidos no se describen en detalle exhaustivo, para evitar ocultar, ocultar u ofuscar innecesariamente las realizaciones de la presente invención.

Sumario

La invención está definida por las reivindicaciones adjuntas y se describe en la sección titulada "Shutter interval messaging in AVC", específicamente en la Tabla 28. Las otras realizaciones y ejemplos se proporcionan con fines ilustrativos, y no representan realizaciones de la invención, excepto cuando se combinan con todas las características definidas respectivamente en las reivindicaciones independientes.

Conducto de procesamiento de entrega de vídeo de ejemplo

La figura 1 representa un proceso de ejemplo de un conducto de entrega de vídeo (100) convencional que muestra diversas etapas desde la captura de vídeo a la visualización de contenido de vídeo. Una secuencia de fotogramas de vídeo (102) se captura o genera usando el bloque de generación de imagen (105). Los fotogramas de vídeo (102) pueden ser capturados digitalmente (por ejemplo, mediante una cámara digital) o generados por un ordenador (por ejemplo, usando animación informática) para proporcionar datos de vídeo (107). Alternativamente, los fotogramas de vídeo (102) pueden ser capturados en película mediante una cámara de película. La película se convierte a un formato digital para proporcionar datos de vídeo (107). En una fase de producción (110), se editan datos de vídeo (107) para proporcionar un flujo de producción (112) de vídeo.

Los datos de vídeo del flujo de producción (112) se proporcionan a continuación a un procesador en el bloque (115) para edición de post-producción. La edición de post-producción de bloque (115) puede incluir ajustar o modificar colores o brillo en áreas particulares de una imagen, para mejorar la calidad de la imagen o lograr una apariencia particular para la imagen de acuerdo con la intención creativa del creador de vídeo. Esto se denomina a veces "temporización del color" o "clasificación del color". Otra edición (por ejemplo, selección y secuenciación de escenas, recorte de imágenes, adición de efectos especiales visuales generados por ordenador, control del temblor o del desenfoque, control de velocidad de fotograma, etc.) puede realizarse en el bloque (115) para producir una versión final (117) de la producción para distribución. Durante la edición de post-producción (115), las imágenes de vídeo se ven en una pantalla de referencia (125). Después de la post-producción (115), los datos de vídeo de la producción final (117) pueden ser entregados al bloque de codificación (120) para entregar en sentido descendente a dispositivos de decodificación y reproducción tales como equipos de televisión, decodificadores, cines y similares. En algunas realizaciones, el bloque de codificación (120) puede incluir codificadores de audio y vídeo, tales como los definidos por ATSC, DVB, DVD, Blu-Ray y otros formatos de entrega, para generar el flujo de bits codificados (122). En un receptor, el flujo de bits codificados (122) es decodificado por la unidad de decodificación (130) para generar una señal decodificada (132) que representa una aproximación idéntica o cercana de la señal (117). El receptor puede estar conectado a una pantalla de objetivo (140) que puede tener características completamente diferentes a la pantalla de referencia (125). En ese caso, se puede usar un bloque de gestión de visualización (135) para mapear el rango dinámico de la señal decodificada (132) a las características de la visualización objetivo (140) generando la señal mapeada en visualización (137).

Codificación escalable

La codificación escalable ya forma parte de una serie de estándares de codificación de vídeo, tales como, MPEG-2, AVC y HEVC. En las realizaciones de esta invención, la codificación escalable se extiende para mejorar el rendimiento y la flexibilidad, especialmente en lo que se refiere a contenido de HDR de muy alta resolución.

Tal como se usa en el presente documento, el término "ángulo de obturador" indica un ajuste de obturador ajustable que controla la proporción de tiempo que la película está expuesta a la luz durante cada intervalo de fotograma. Por ejemplo, en una realización

$$\frac{\text{ángulo de obturador}}{360} = \frac{\text{tiempo de exposición}}{\text{intervalo de fotograma}} \quad (1)$$

El término proviene de obturadores giratorios mecánicos, heredados; sin embargo, las cámaras digitales modernas también pueden ajustar su obturador electrónicamente. Los cinematógrafos pueden usar el ángulo del obturador para controlar la cantidad de desenfoque o temblor por movimiento que se graba en cada fotograma. Obsérvese que en lugar de usar "tiempo de exposición" también se pueden usar términos alternativos, tales como "duración de la exposición", "intervalo del obturador" y "velocidad del obturador". De manera similar, en lugar de usar "intervalo de

fotograma” se puede usar el término “duración de fotograma”. Alternativamente, se puede sustituir el “intervalo de fotograma” por “1/velocidad de fotograma”. El valor del tiempo de exposición es típicamente menor o igual que la duración de un fotograma. Por ejemplo, un ángulo de obturador de 180 grados indica que el tiempo de exposición es la mitad de la duración del fotograma. En algunas situaciones, el tiempo de exposición puede ser mayor que la duración de fotograma del vídeo codificado, por ejemplo, cuando la velocidad de fotograma codificada es de 120 fps y la velocidad de fotograma del contenido de vídeo asociado antes de la codificación y visualización es de 60 fps.

Considérese, sin limitación, una realización en la que el contenido original se dispara (o se genera) a una velocidad de fotograma original (por ejemplo, 120 fps) con un ángulo de obturador de 360 grados. A continuación, en un dispositivo receptor, se puede renderizar la salida de vídeo a una variedad de velocidades de fotograma iguales o inferiores a la velocidad de fotograma original mediante una combinación crítica de los fotogramas originales, por ejemplo, promediando u otras operaciones conocidas en la técnica.

El proceso de combinación puede realizarse con señales codificadas no lineales (por ejemplo, usando gamma, PQ o HLG), pero la mejor calidad de imagen se obtiene combinando fotogramas en el dominio de la luz lineal, convirtiendo en primer lugar las señales codificadas no lineales en representaciones de luz lineal, a continuación, combinando los fotogramas convertidos y finalmente recodificando la salida con la función de transferencia no lineal. Este proceso proporciona una simulación más precisa de una exposición física de cámara que la combinación en el dominio no lineal.

En términos generales, el proceso de combinación de fotogramas puede expresarse en términos de la velocidad de fotograma original, la velocidad de fotograma objetivo, el ángulo de obturador objetivo y el número de fotogramas que van a combinarse como:

$$n_frames = \{target_shutter_angle/360\} * (original_frame_rate/target_frame_rate), \quad (2)$$

que es equivalente a

$$target_shutter_angle = 360 * n_frames * (target_frame_rate/original_frame_rate), \quad (3)$$

donde n_frames es el número de fotogramas combinados, $original_frame_rate$ es la velocidad de fotograma del contenido original, $target_frame_rate$ es la velocidad de fotograma que se va a renderizar (donde, $target_frame_rate \leq original_frame_rate$), y $target_shutter_angle$ indica la cantidad de desenfoque por movimiento deseada. En este ejemplo, el valor máximo de $target_shutter_angle$ es de 360 grados y corresponde al desenfoque por movimiento máximo. El valor mínimo de $target_shutter_angle$ puede expresarse como $360 * (target_frame_rate/original_frame_rate)$ y corresponde a un desenfoque por movimiento mínimo. El valor máximo de n_frames puede expresarse como $(original_frame_rate/target_frame_rate)$. Los valores de $target_frame_rate$ y $target_shutter_angle$ deben seleccionarse de tal manera que el valor de n_frames sea un número entero distinto de cero.

En el caso especial de que la velocidad de fotograma original sea de 120 fps, la ecuación (2) puede reescribirse como

$$n_frames = target_shutter_angle / (3 * target_frame_rate), \quad (4)$$

que es equivalente a

$$target_shutter_angle = 3 * n_frames * target_frame_rate. \quad (5)$$

Las relaciones entre los valores de $target_frame_rate$, n_frames y $target_shutter_angle$ se muestran en la Tabla 1 para el caso de $original_frame_rate = 120$ fps. En la Tabla 1, “NA” indica que no se permite la combinación correspondiente de una velocidad de fotograma objetivo y el número de fotogramas combinados.

Tabla 1: Relación entre la velocidad de fotograma objetivo, el número de fotogramas combinados y el ángulo de obturador objetivo, para una velocidad de fotograma original de 120 fps.

Velocidad de fotograma objetivo (fps)	Número de fotogramas combinados				
	5	4	3	2	1
	Ángulo de obturador objetivo (grados)				
24	360	288	216	144	72
30	NA	360	270	180	90

40	NA	NA	360	240	120
60	NA	NA	NA	360	180

La figura 2 representa un proceso de ejemplo de combinación de fotogramas originales consecutivos para renderizar una velocidad de fotograma objetivo en un ángulo de obturador objetivo de acuerdo con una realización. Dada una secuencia de entrada (205) a 120 fps y un ángulo de obturador de 360 grados, el proceso genera una secuencia de vídeo de salida (210) a 24 fps y un ángulo de obturador de 216 grados combinando tres de los fotogramas de entrada en un conjunto de cinco fotogramas consecutivos (por ejemplo, los primeros tres fotogramas consecutivos), y descartando los otros dos. Obsérvese que en algunas realizaciones, el fotograma-01 de salida (210) puede generarse combinando fotogramas (205) de entrada alternativos, tales como los fotogramas 1, 3 y 5, o los fotogramas 2, 4 y 5, y similares; sin embargo, lo esperado es que combinar fotogramas consecutivos producirá una salida de vídeo de mejor calidad.

Es deseable soportar contenido original con velocidad de fotograma variable, por ejemplo, para gestionar el efecto artístico y estilístico. También es deseable que la velocidad de fotograma de entrada variable del contenido original se empaquete en un "contenedor" que tiene una velocidad de fotograma fija, para simplificar la producción, intercambio y distribución del contenido. Como ejemplo, se presentan tres realizaciones sobre cómo representar los datos de vídeo de velocidad de fotograma variable en un contenedor de velocidad de fotograma fija. Con fines de claridad y sin limitación, las siguientes descripciones usan un contenedor fijo de 120 fps, pero los enfoques pueden extenderse fácilmente a un contenedor de velocidad de fotograma alternante.

Primera realización (Velocidad de fotograma variable)

La primera realización es una descripción explícita del contenido original que tiene una velocidad de fotograma variable (no constante) empaquetado en un contenedor que tiene una velocidad de fotograma constante. Por ejemplo, el contenido original que tiene una velocidad de fotograma diferente, digamos, de 24, 30, 40, 60 o 120 fps, para diferentes escenas, puede empaquetarse en un contenedor que tiene una velocidad de fotograma constante de 120 fps. Para este ejemplo, cada fotograma de entrada puede duplicarse 5x, 4x, 3x, 2x o 1x veces para empaquetarlo en un contenedor común de 120 fps.

La figura 3 representa un ejemplo de una secuencia A de vídeo de entrada con velocidad de fotograma variable y ángulo de obturador variable que se representa en un flujo de bits codificado B con una velocidad de fotograma fija. A continuación, en un decodificador, el decodificador reconstruye la secuencia C de vídeo de salida a la velocidad de fotograma deseada y el ángulo de obturador, que puede cambiar de escena a escena. Por ejemplo, tal como se representa en la figura 3, para construir la secuencia B, algunos de los fotogramas de entrada se duplican, algunos se codifican como están (sin duplicación), y algunos se copian cuatro veces. A continuación, para construir la secuencia C, se selecciona cualquier fotograma de cada conjunto de fotogramas duplicados para generar fotogramas de salida, que coincidan con la velocidad de fotograma original y el ángulo de obturador.

En esta realización, los metadatos se insertan en el flujo de bits para indicar la velocidad de fotograma original (base) y el ángulo de obturador. Los metadatos pueden señalizarse usando sintaxis de alto nivel tal como un conjunto de parámetros de secuencia (Sequence Parameter Set, SPS), un conjunto de parámetros de imagen (Picture Parameter Set, PPS), una cabecera de segmento o grupo de mosaicos y similares. La presencia de metadatos permite a los codificadores y decodificadores realizar funciones beneficiosas, tales como:

a) Un codificador puede ignorar fotogramas duplicados, aumentando de este modo la velocidad de codificación y simplificando el procesamiento. Por ejemplo, todas las unidades de árbol de codificación (Coding Tree Units, CTU) en fotogramas duplicados pueden codificarse usando el modo SALTAR y el índice de referencia 0 en la LISTA 0 de los fotogramas de referencia, que se refiere a un fotograma decodificado a partir del cual se copian los fotogramas duplicados.

b) Un decodificador puede evitar la decodificación de fotogramas duplicados simplificando de este modo el procesamiento. Por ejemplo, los metadatos en el flujo de bits pueden indicar que un fotograma es un duplicado de un fotograma decodificado previamente que el decodificador puede reproducir copiando y sin decodificar el nuevo fotograma.

c) Un dispositivo de reproducción puede optimizar el procesamiento más abajo indicando la velocidad de fotograma base, por ejemplo ajustando la conversión de velocidad de fotograma o los algoritmos de reducción de ruido.

Esta realización permite a un usuario final ver el contenido renderizado a las velocidades de fotograma previstas por los creadores de contenido. Esta realización no proporciona compatibilidad con versiones anteriores, con dispositivos que no soportan la velocidad de fotograma del contenedor, por ejemplo, 120 fps.

Las tablas 2 y 3 representan sintaxis de ejemplo de carga útil de secuencia de bytes sin procesar (Raw Byte Sequence Payload, RBSP) para un conjunto de parámetros de secuencia y cabecera de grupo de mosaicos, donde los nuevos elementos sintácticos propuestos se representan en una *fuentes cursiva*. La sintaxis restante sigue la sintaxis en la especificación propuesta del códec de vídeo versátil (VVC) (ref. [2]).

- 5 Como ejemplo, en SPS (véase la Tabla 2), se puede añadir un indicador para habilitar la velocidad de fotograma variable.

sps_vfr_enabled_flag igual a 1 especifica que la secuencia de vídeo codificada (Coded Video Sequence, CVS) puede contener contenido de velocidad de fotograma variable, **sps_vfr_enabled_flag** igual a 0 especifica que la CVS contiene contenido de velocidad de fotograma fija.

- 10 En la cabecera() de tile_group (véase la Tabla 3),

tile_group_vrf_info_present_flag igual a 1 especifica que los elementos sintácticos **tile_group_true_fr** y **tile_group_shutterangle** están presentes en la sintaxis. **tile_group_vrf_info_present_flag** igual a 0 especifica que los elementos sintácticos **tile_group_true_fr** y **tile_group_shutterangle** no están presentes en la sintaxis. Cuando **tile_group_vrf_info_present_flag** no está presente, se deduce que es 0.

- 15 **tile_group_true_fr** indica la velocidad de fotograma verdadera de los datos de vídeo transportados en este flujo de bits. **tile_group_shutterangle** indica el ángulo de obturador correspondiente a la velocidad de fotograma verdadera de los datos de vídeo transportados en este flujo de bits.

tile_group_skip_flag igual a 1 especifica que el grupo de mosaicos actual se copia de otro grupo de mosaicos. **tile_group_skip_flag** igual a 0 especifica que el grupo de mosaicos actual no se copia de otro grupo de mosaicos.

- 20 **tile_group_copy_pic_order_cnt_lsb** especifica el módulo de recuento de orden de imagen MaxPicOrderCntLsb para la imagen decodificada previamente que la imagen actual copia desde cuando **tile_group_skip_flag** se ajusta en 1.

Tabla 2: Sintaxis de RBSP de conjunto de parámetros de ejemplo para contenido con velocidad de fotograma variable

seq_parameter_set_rbsp() {	Descriptor
sps_max_sub_layers_minus1	u(3)
sps_reserved_zero_5bits	u(5)
profile_tier_level(sps_max_sub_layers_minus1)	
sps_seq_parameter_set_id	uc(v)
...	
<i>sps_vfr_enabled_flag</i>	<i>u(1)</i>
...	
sps_extension_flag	u(1)
if(sps_extension_flag)	
while(more_rbsp_data())	
sps_extension_data_flag	u(1)
rbsp_trailing_bits()	
}	

Tabla 3: Ejemplo de sintaxis de cabecera de grupo de mosaicos con soporte para contenido con velocidad de fotograma variable

<code>tile_group_header() {</code>	Descriptor
<code> tile_group_pic_parameter_set_id</code>	<code>ue(v)</code>
<code> if(NumTilesInPic > 1) {</code>	
<code> tile_group_address</code>	<code>u(v)</code>
<code> num_tiles_in_tile_group_minus1</code>	<code>ue(v)</code>
<code> }</code>	
<code> tile_group_type</code>	<code>ue(v)</code>
<code> tile_group_pic_order_cnt_lsb</code>	<code>u(v)</code>
<code> if(sps_vfr_enabled_flag) {</code>	
<code> tile_group_vfr_info_present_flag</code>	<code>u(1)</code>
<code> if(tile_group_vfr_info_present_flag) {</code>	
<code> tile_group_true_fr</code>	<code>u(9)</code>
<code> tile_group_shutterangle</code>	<code>u(9)</code>
<code> }</code>	
<code> tile_group_skip_flag</code>	<code>u(1)</code>
<code> }</code>	
<code> if(tile_group_skip_flag)</code>	
<code> tile_group_copy_pic_order_cnt_lsb</code>	<code>u(v)</code>
<code> } else {</code>	
<code> ALL OTHER TILE_GROUP_SYNTAX</code>	
<code> }</code>	
<code> if(num_tiles_in_tile_group_minus1 > 0) {</code>	
<code> offset_len_minus1</code>	<code>ue(v)</code>
<code> for(i = 0; i < num_tiles_in_tile_group_minus1; i++)</code>	
<code> entry_point_offset_minus1[i]</code>	<code>u(v)</code>
<code> }</code>	
<code> byte_alignment()</code>	
<code>}</code>	

Segunda realización - Contenedor de velocidad de fotograma fija

La segunda realización permite el caso de uso en el que el contenido original que tiene una velocidad de fotograma y un ángulo de obturador fijos puede renderizarse mediante un decodificador a una velocidad de fotograma alternante y a un ángulo de obturador simulado variable, tal como se ilustra en la figura 2. Por ejemplo, en el caso de que el contenido original tenga una velocidad de fotograma de 120 fps y un ángulo de obturador de 360 grados (lo que significa que el obturador está abierto 1/120 segundos), un decodificador puede renderizar múltiples velocidades de fotograma que son menores o iguales a 120 fps. Por ejemplo, tal como se describe en la Tabla 1, para decodificar 24 fps con un ángulo de obturador simulado de 216 grados, el decodificador puede combinar tres fotogramas decodificados y mostrar a 24 fps. La Tabla 4 se expande sobre la Tabla 1 e ilustra cómo combinar diferentes números de fotogramas codificados para renderizar a las velocidades de fotograma objetivo de salida y los ángulos de obturador objetivo deseados. La combinación de los fotogramas puede realizarse mediante una media de píxeles simple, mediante una media de píxeles ponderada, donde los píxeles de un determinado fotograma pueden ponderarse más que los píxeles de otros fotogramas, y la suma de todas las sumas de pesos a uno, o mediante otros esquemas de interpolación de filtro conocidos en la técnica. En la Tabla 4, la función $Ce(a,b)$ denota la combinación de fotogramas codificados a a b, donde la combinación puede realizarse promediando, promediando ponderadamente, filtrando y similares.

Tabla 4: Ejemplo de combinación de fotogramas de entrada a 120 fps para generar fotogramas de salida a fps objetivo y valores de ángulo de obturador

Entrada	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10
Cod.	e1	e2	e3	e4	e5	e6	e7	e8	e9	e10
Dec. 120fps										
@360	e1	e2	e3	e4	e5	e6	e7	e8	e9	e10
Dec. 60fps										
@360	Ce(1,2)		Ce(3,4)		Ce(5,6)		Ce(7,8)		Ce(9,10)	
@180	e1		e3		e5		e7		e9	
Dec. 40fps										
@360	Ce(1,3)			Ce(4,6)			Ce(7,9)			Ce(10,12)
@240	Ce(1,2)			Ce(4,5)			Ce(7,8)			Ce(10,11)
@120	e1			e4			e7			e10
Dec. 30fps										
@360	Ce(1,4)				Ce(5,8)				Ce(9,12)	
@270	C(1,3)				Ce(5,7)				Ce(9,11)	
@180	Ce(1,2)				Ce(5,6)				Ce(9,10)	
@90	e1				e5				e9	
Dec. 24fps										

@360	Ce(1,5)	Ce(6,10)
@288	Ce(1,4)	Ce(6,9)
@216	Ce(1,3)	Ce(6,8)
@144	Ce(1,2)	Ce(6,7)
@72	e1	e6

Cuando el valor del ángulo de obturador objetivo es menor de 360 grados, el decodificador puede combinar diferentes conjuntos de fotogramas decodificados. Por ejemplo, a partir de la Tabla 1, dado un flujo original de 120 fps @ 360 grados, para generar un flujo a 40 fps y un ángulo de obturador de 240 grados, un decodificador necesita combinar dos fotogramas de entre tres fotogramas posibles. Por lo tanto, puede combinar el primer y el segundo fotograma o el segundo y el tercer fotograma. La elección de qué fotogramas combinar puede describirse en términos de una "fase de decodificación" expresada como:

$$\text{decode_phase} = \text{decode_phase_idx} * (360 / n_frames), \quad (6)$$

donde `decode_phase_idx` indica el índice de desplazamiento dentro de un conjunto de fotogramas secuenciales que tienen valores de índice en $[0, n_frames_max - 1]$, donde `n_frames` viene dado por la ecuación (2), y

$$n_frames_max = \text{orig_frame_rate} / \text{target_frame_rate}. \quad (7)$$

En general, `decode_phase_idx` varía en el intervalo $[0, n_frames_max - n_frames]$. Por ejemplo, para una secuencia original a 120 fps y un ángulo de obturador de 360 grados, para la velocidad de fotograma objetivo de 40 fps a un ángulo de obturador de 240 grados, $n_frames_max = 120 / 40 = 3$. A partir de la ecuación (2), $n_frames = 2$, por lo tanto, `decode_phase_idx` varía en el intervalo $[0, 1]$. Por lo tanto, `decode_phase_idx = 0` indica seleccionar fotogramas con índice 0 y 1, y `decode_phase_idx = 1` indica seleccionar fotogramas con índice 1 y 2.

En esta realización, la velocidad de fotograma variable renderizada prevista por el creador de contenido puede señalizarse como metadatos, tal como un mensaje de información de mejora suplementaria (Supplemental Enhancement Information, SEI) o como información de usabilidad de vídeo (Video Usability Information, VUI). Opcionalmente, la velocidad de fotograma renderizada puede ser controlada por el receptor o por un usuario. Un ejemplo de mensajería de SEI de conversión de velocidad de fotograma que especifica la velocidad de fotograma preferida y el ángulo de obturador del creador de contenido, se muestra en la Tabla 5. El mensaje de SEI también puede indicar si se realizan fotogramas de combinación en el dominio de la señal codificada (por ejemplo, gamma, PQ, etc.) o el dominio de la luz lineal. Obsérvese que el post-procesamiento requiere una memoria intermedia de fotogramas además de la memoria intermedia de imágenes de decodificador (Decoder Picture Buffer, DPB). El mensaje de SEI puede indicar cuántas memorias intermedias de fotogramas adicionales se necesitan, o algún método alternativo para combinar fotogramas. Por ejemplo, para reducir la complejidad, los fotogramas pueden recombinarse a resolución espacial reducida.

Tal como se representa en la Tabla 4, a ciertas combinaciones de velocidades de fotograma y ángulos de obturador (por ejemplo, a 30 fps y 360 grados o a 24 fps y 288 o 360 grados) un decodificador puede necesitar combinar más de tres fotogramas decodificados, lo que aumenta la cantidad de espacio de memoria intermedia requerido por el decodificador. Para reducir la carga de espacio de memoria intermedia adicional en el decodificador, en algunas realizaciones, ciertas combinaciones de velocidades de fotograma y ángulos de obturador pueden estar fuera de los límites del conjunto de parámetros de decodificación permitidos (por ejemplo, ajustando perfiles y niveles de codificación apropiados).

Considerando de nuevo, como ejemplo, el caso de reproducción a 24 fps, un decodificador puede decidir visualizar el mismo fotograma cinco veces para ser visualizado a una velocidad de fotograma de salida de 120 fps. Esto es exactamente lo mismo que mostrar el fotograma una sola vez a una velocidad de fotograma de salida de 24 fps. La ventaja de mantener una velocidad de fotograma de salida constante es que una visualización puede funcionar a una velocidad de reloj constante, lo que hace que todo el hardware sea mucho más simple. Si la pantalla puede variar dinámicamente la velocidad de reloj, entonces puede tener más sentido mostrar el fotograma solamente una vez (para 1/24 de un segundo), en lugar de repetir el mismo fotograma cinco veces (cada 1/120 de un segundo). El primer enfoque puede dar como resultado una calidad de imagen ligeramente mayor, una mayor eficiencia óptica o una mayor eficiencia energética. Consideraciones similares son también aplicables a otras velocidades de fotograma.

La Tabla 5 representa un ejemplo de una sintaxis de mensajería de SEI de conversión de velocidad de fotograma de acuerdo con una realización.

Tabla 5: Ejemplo de sintaxis de mensaje de SEI que permite la conversión de velocidad de fotograma

framerate_conversion(payloadSize) {	Descriptor
framerate_conversion_cancel_flag	u(1)
if(!frame_conversion_cancel_flag) {	
base_frame_rate	u(9)
base_shutter_angle	u(9)
decode_phase_idx_present_flag	u(1)
if (decode_phase_idx_present_flag) {	
decode_phase_idx	u(3)
}	
conversion_domain_idc	u(1)
num_frame_buffer	u(3)
framerate_conversion_persistence_flag	u(1)
}	
}	

framerate_conversion_cancel_flag igual a 1 indica que el mensaje de SEI cancela la persistencia de cualquier mensaje de SEI de conversión de velocidad de fotograma anterior en orden de salida.
framerate_conversion_cancel_flag igual a 0 indica que sigue la información de conversión de velocidad de fotograma.

base_frame_rate especifica la velocidad de fotograma deseada.

base_shutter_angle especifica el ángulo de obturador deseado.

decode_phase_idx_present_flag igual a 1 especifica que la información de fase de decodificación está presente.
decode_phase_idx_present_flag igual a 0 especifica que la información de fase de decodificación no está presente.

decode_phase_idx indica el índice de desplazamiento dentro de un conjunto de fotogramas secuenciales que tienen valores de índice en el intervalo 0 .. (n_frames_max-1) donde n_frames_max = 120/base_frame_rate. El valor de **decode_phase_idx** estará en el intervalo 0 .. (n_frames_max-n_frames), donde n_frames = base_shutter_angle/(3*base_frame_rate). Cuando **decode_phase_idx** no está presente, se deduce que es 0.

conversion_domain_idc igual a 0 especifica que la combinación de fotogramas se realiza en el dominio lineal.
conversion_domain_idc igual a 1 especifica que la combinación de fotogramas se realiza en el dominio no lineal.

num_frame_buffers especifica el número adicional de memorias intermedias de fotograma (sin contar la DPB).

framerate_conversion_persistence_flag especifica la persistencia del mensaje de SEI de conversión de velocidad de fotograma para la capa actual. **framerate_conversion_persistence_flag** igual a 0 especifica que el mensaje de SEI de conversión de velocidad de fotograma se aplica solamente a la imagen decodificada actual. Sea picA la imagen actual. **framerate_conversion_persistence_flag** igual a 1 especifica que el mensaje de SEI de conversión de velocidad de fotograma persiste para la capa actual en orden de salida hasta que una o más de las siguientes condiciones son verdaderas:

- Comienza una nueva secuencia de vídeo codificada por capas (Coded Layer-wise Video Sequence, CLVS) de la capa actual.

- El flujo de bits finaliza.
- Una imagen picB en la capa actual en una unidad de acceso que contiene un mensaje de SEI de conversión de velocidad de fotograma que es aplicable a la capa actual para la que $\text{PicOrderCnt}(\text{picB})$ es mayor que $\text{PicOrderCnt}(\text{picA})$, donde $\text{PicOrderCnt}(\text{picB})$ y $\text{PicOrderCnt}(\text{picA})$ son los valores de PicOrderCntVal de picB y picA, respectivamente, inmediatamente después de la invocación del proceso de decodificación para el recuento de orden de imagen para picB.

Tercera realización - Entrada codificada en múltiples ángulos de obturador

Una tercera realización es un esquema de codificación que permite la extracción de velocidades de subtrama del flujo de bits, soportando así la compatibilidad con versiones anteriores. En la HEVC, esto se logra mediante la escalabilidad temporal. La escalabilidad de capa temporal se habilita asignando diferentes valores a un elemento sintáctico temporal_id para los fotogramas decodificados. El flujo de bits puede extraerse de este modo simplemente sobre la base de los valores de temporal_id. Sin embargo, el enfoque de estilo de HEVC para la escalabilidad temporal no permite la renderización de velocidades de fotograma de salida con diferentes ángulos de obturador. Por ejemplo, una velocidad de fotograma base de 60 fps extraída de una original de 120 fps siempre tendrá un ángulo de obturador de 180 grados.

En ATSC 3.0, se describe un método alternativo en el que los fotogramas a 60 fps que tienen ángulos de obturador de 360 grados se emulan como una media ponderada de dos fotogramas de 120 fps. A los fotogramas emulados de 60 fps se les asigna un valor de temporal_id de 0 y se combinan con fotogramas originales alternantes de 120 fps a los que se les asigna un valor de temporal_id de 1. Cuando se necesitan 60 fps, el decodificador solo necesita decodificar fotogramas con un temporal_id de 0. Cuando se necesitan 120 fps, el decodificador puede restar cada fotograma con temporal_id = 1 (es decir, un fotograma de 120 fps), de una versión escalada de cada fotograma con temporal_id = 0 correspondiente (es decir, fotograma emulado de 60 fps) para recuperar el fotograma original correspondiente de 120 fps que no se transmitió explícitamente, reconstituyendo de este modo todas los fotogramas originales de 120 fps.

En realizaciones de esta invención, se describe un nuevo algoritmo que soporta múltiples velocidades de fotograma objetivo y ángulos de obturador objetivo de una manera que es compatible con versiones anteriores (Backward Compatible, BC). La propuesta es preprocesar el contenido original de 120 fps a una velocidad de fotograma base en varios ángulos de obturador. Entonces, en el decodificador, pueden deducirse fácilmente otras velocidades de fotograma en diversos otros ángulos de obturador. El enfoque de ATSC 3.0 puede considerarse como un caso especial del esquema propuesto, donde los fotogramas con temporal_id = 0 transportan fotogramas en un ángulo de obturador de 60fps@360, y los fotogramas con temporal_id = 1 transportan fotogramas en un ángulo de obturador de 60fps@180.

Como primer ejemplo, tal como se representa en la figura 4, considérese una secuencia de entrada a 120 fps y un ángulo de obturador de 360 que se usa para codificar una secuencia con una velocidad de fotograma de capa base de 40 fps y ángulos de obturador de 120, 240 y 360 grados. En este esquema, el codificador calcula nuevos fotogramas combinando hasta tres de los fotogramas de entrada originales. Por ejemplo, el fotograma 2 codificado (En-2) que representa la entrada a 40 fps y 240 grados se genera combinando los fotogramas de entrada 1 y 2, y el fotograma 3 codificado (En-3) que representa la entrada a 40 fps y 360 grados se genera combinando el fotograma En-2 con el fotograma 3 de entrada. En el decodificador, para reconstruir la secuencia de entrada, el fotograma 2 decodificado (Dec-2) se genera restando el fotograma En-1 del fotograma En-2, y el fotograma 3 decodificado (Dec-3) se genera restando el fotograma En-2 del fotograma En-3. Los tres fotogramas decodificados representan una salida a una velocidad de fotograma base de 120 fps y un ángulo de obturador de 360 grados. Velocidades de fotograma y ángulos de obturador adicionales pueden ser extrapolados utilizando los fotogramas decodificados tal como se representan en la Tabla 6. En la Tabla 6, la función $Cs(a,b)$ denota la combinación de fotogramas de entrada a a b, donde la combinación puede realizarse promediando, promediando ponderadamente, filtrando y similares.

Tabla 6: Ejemplo de combinación de fotogramas con una línea de referencia de 40 fps

Tramas de entrada 120fps @360	s1	s2	s3	s4	s5	s6	s7	s8	s9
Tramas codificadas 120fps	e1 = s1	e2 = Cs(1,2)	e3 = Cs(1,3)	e4 = s4	e5 = Cs(4, 5)	e6 = Cs(4,6)	e7 = s7	e8 = Cs(7, 8)	e9 = Cs(7, 9)

ES 3 009 731 T3

Decodificar 120fps									
@360	$e_1 = s_1$	$e_2 - e_1 = s_2$	$e_3 - e_2 = s_3$	$e_4 = s_4$	$e_5 - e_4 = s_5$	$e_6 - e_4 = s_6$	$e_7 = s_7$	$e_8 - e_7 = s_8$	$e_9 - e_8 = s_9$
Decodificar 60fps									
@360	e_2	$e_3 - e_2 + e_4 = Cs(3,4)$			$e_6 - e_4 = Cs(5,6)$		$e_8 = Cs(7,8)$		$e_9 - e_8 + e_{10}$
@180	e_1	$e_3 - e_2 = s_3$		$e_5 - e_4 = s_5$		e_7		$e_9 - e_8$	
Decodificar 40fps									
@360	$e_3 = Cs(1,3)$			e_6		e_9			
@240	$e_2 = Cs(1,2)$			e_5		e_8			
@120	$e_1 = s_1$			e_4		e_7			
Decodificar 30fps									
@360	$e_3 + e_4 = Cs(1,4)$				$e_6 - e_4 + e_8 = Cs(5,8)$			$e_9 - e_8 + e_{12}$	
@270	$e_3 = Cs(1,3)$				$e_6 - e_5 + e_7 = Cs(5,7)$			$e_9 - e_8 + e_{11}$	
@180	$e_2 = Cs(1,2)$				$e_6 - e_4 = Cs(5,6)$			$e_9 - e_8 + e_{10}$	
@90	e_1				$e_5 - e_4 = s_5$			$e_9 - e_8$	
Decodificar 24fps									
@360	$e_3 + e_5 = Cs(1,5)$					$e_6 - e_5 + e_9 + e_{10} = Cs(6,10)$			
@288	$e_3 + e_4 = Cs(1,4)$					$e_6 - e_5 + e_9 = Cs(6,9)$			
@216	$e_3 = Cs(1,3)$					$e_6 - e_5 + e_8 = Cs(6,8)$			
@144	$e_2 = Cs(1,2)$					$e_6 - e_5 + e_7 = Cs(6,7)$			
@72	$e_1 = s_1$					$e_6 - e_5 = s_6$			

Una ventaja de este enfoque es que, tal como se representa en la Tabla 6, todas las versiones de 40 fps pueden ser decodificadas sin ningún procesamiento adicional. Otra ventaja es que se pueden deducir otras velocidades de fotograma en diversos ángulos de obturador. Por ejemplo, considérese una decodificación de decodificador a 30 fps y un ángulo de obturador de 360. A partir de la Tabla 4, la salida corresponde a la secuencia de fotogramas generados por $Ce(1,4) = Cs(1,4)$, $Cs(5,8)$, $Cs(9,12)$ y similares, que coincide también con la secuencia de decodificación representada en la Tabla 6; sin embargo, en la Tabla 6, $Cs(5,8) = e6-e4+e8$. En una realización, se pueden usar tablas de consulta (Look-Up Tables, LUT) para definir cómo necesitan ser combinados los fotogramas decodificados para generar una secuencia de salida a la velocidad de fotograma de salida especificada y el ángulo de obturador emulado.

En otro ejemplo, se propone combinar hasta cinco fotogramas en el codificador con el fin de simplificar la extracción de la capa base de 24 fps en ángulos de obturador de 72, 144, 216, 288 y 360 grados, tal como se muestra a continuación. Esto es deseable para contenido de película que se presente mejor en 24 fps en televisiones heredadas.

Tabla 7: Ejemplo de combinación de fotograma con una línea de referencia de 24 fps

Fotogramas de entrada 120 fps @360	s1	s2	s3	s4	s5	s6	s7	s8	s9
Cod. fotogramas	$e1 = s1$	$e2 = Cs(1,2)$	$e3 = Cs(1,3)$	$e4 = Cs(1,4)$	$e5 = Cs(1,5)$	$e6 = s6$	$e7 = Cs(6,7)$	$e8 = Cs(6,8)$	$e9 = Cs(6,9)$
Decodificar 120 fps									
@360	e1	e2-e1	e3-e2	e4-e3	e5-e4	e6	e7-e6	e8-e7	e9-e8
Decodificar 60 fps									
@360	e2		e4-e2		e5-e4+e6		e8-e6		e10-e8
@180	e1		e3-e2		e5-e4		e7-e6		e9-e8
Decodificar 40 fps									
@360	e3			e5-e3+e6			e9-e6		
@240	e2			e5-e3			e8-e6		
@120	e1			e4-e3			e7-e6		
Decodificar 30fps									
@360	e4				e5-e4+e8				e10-e8+e12
@270	e3				e5-e4+e7				e10-e8+e11

@180	e2	e5-e4+e6				e10-e8			
@90	e1	e5-e4				e9-e8			
Decodificar 24fps									
@360	e5	e10							
@288	e4	e9							
@216	e3	e8							
@144	e2				e7				
@72	e1				e6				

Tal como se representa en la Tabla 7, si la velocidad de fotograma de decodificación coincide con la velocidad de fotograma de referencia (24 fps), entonces, en cada grupo de cinco fotogramas (por ejemplo, e1 a e5) un decodificador puede seleccionar simplemente el un fotograma en el ángulo de obturador deseado (por ejemplo, e2 para un ángulo de obturador en 144 grados). Para decodificar a una velocidad de fotograma diferente y un ángulo de obturador específico, el decodificador necesitará determinar cómo combinar adecuadamente (por ejemplo, mediante adición o sustracción) los fotogramas decodificados. Por ejemplo, para decodificar a 30 fps y un ángulo de obturador de 180 grados, pueden seguirse las siguientes etapas:

a) El decodificador puede considerar un codificador hipotético que transmite a 120 fps y 360 grados sin ninguna consideración para compatibilidad con versiones anteriores; entonces, a partir de la Tabla 1, el decodificador necesita combinar 2 de 4 fotogramas para generar la secuencia de salida a la velocidad de fotograma y el ángulo de obturador deseados. Por ejemplo, tal como se representa en la Tabla 4, la secuencia incluye, $Ce(1,2) = \text{Avg}(s1,s2)$, $Ce(5,6) = \text{Avg}(s5,s6)$ y similares, donde $\text{Avg}(s1,s2)$ puede indicar la media de los fotogramas s1 y s2.

b) Dado que, por definición, los fotogramas codificados pueden expresarse como $e1 = s1$, $e2 = \text{Avg}(s1,s2)$, $e3 = \text{Avg}(s1,s3)$, y similares, puede deducirse fácilmente que la secuencia de fotogramas en la etapa a) puede expresarse también como:

$$Ce(1,2) = \text{Avg}(s1,s2) = e2$$

$$Ce(5,6) = \text{Avg}(s5,s6) = \text{Avg}(s1,s5) - \text{Avg}(s1,s4) + s6 = e5 - e4 + e6$$

etc.

Como antes, la combinación apropiada de fotogramas decodificados puede calcularse previamente y estar disponible como una LUT.

Una ventaja del método propuesto es que proporciona opciones tanto para creadores de contenido como para usuarios; es decir, permite la elección directora/editorial y la elección del usuario. Por ejemplo, el preprocesamiento de contenido en el codificador permite crear una velocidad de fotograma de referencia con varios ángulos de obturador. A cada ángulo de obturador se le puede asignar un valor temporal_id en el intervalo $[0, (n_frames - 1)]$, donde n_frames tiene un valor igual a 120 dividido por la velocidad de fotograma base. (Por ejemplo, para una velocidad de fotograma de referencia de 24 fps, temporal_id está en el intervalo $[0, 4]$). La elección puede hacerse para optimizar la eficiencia de compresión, o por razones estéticas. En algunos casos de uso, digamos, para la transmisión en continuo superior, múltiples flujos de bits con diferentes capas base pueden codificarse y almacenarse y ser ofrecidos a los usuarios para seleccionarlos.

En un segundo ejemplo de los métodos divulgados, se pueden soportar múltiples velocidades de fotograma compatibles con versiones anteriores. Idealmente, se puede desear poder decodificar a 24 fotogramas por segundo para obtener una capa base de 24 fps, a 30 fotogramas por segundo para obtener una secuencia de 30 fps, a 60 fotogramas por segundo para obtener una secuencia de 60 fps, y similares. Si no se especifica un ángulo objetivo de obturador, se recomienda un ángulo objetivo de obturador por defecto, entre aquellos ángulos de obturador permisibles para las velocidades de fotograma original y objetivo, lo más cercano posible a 180 grados. Por ejemplo, para los

valores representados en la Tabla 7, los ángulos de obturador objetivo preferidos para fps a 120, 60, 40, 30 y 24 son 360, 180, 120, 180 y 216 grados.

5 A partir de los ejemplos anteriores puede observarse que la elección de cómo codificar el contenido puede influir en la complejidad de decodificar velocidades de fotogramas de capa base específicas. Una realización de esta invención es elegir adaptativamente el esquema de codificación basándose en la velocidad de fotograma de capa base deseada. Para contenido de película esta puede ser de 24 fps, por ejemplo, mientras que para deportes puede ser de 60 fps.

La sintaxis de ejemplo para la realización de BC de la presente invención se muestra a continuación y en las Tablas 8 y 9.

10 En SPS (Tabla 8), se añaden dos elementos sintácticos: `sps_hfr_BC_enabled_flag` y `sps_base_framerate` (si `sps_hfr_BC_enabled_flag` se establece igual a 1).

sps_hfr_BC_enabled_flag igual a 1 especifica que la alta velocidad de fotograma con compatibilidad con versiones anteriores está habilitada en la secuencia de vídeo codificada (CVS). `sps_hfr_BC_enabled_flag` igual a 0 especifica que la alta velocidad de fotograma con compatibilidad con versiones anteriores no está habilitada en la CVS.

sps_base_framerate especifica la velocidad de fotograma de referencia para la CVS actual.

15 En la cabecera del grupo de mosaicos, si `sps_hfr_BC_enabled_flag` se establece en 1, el `number_avg_frames` sintáctico se envía en el flujo de bits.

number_avg_frames especifica el número de fotogramas en la velocidad de fotograma más alta (por ejemplo, 120 fps) que se combinan para generar la imagen actual en la velocidad de fotograma de referencia.

Tabla 8: Sintaxis de RBSP de ejemplo para entrada en diversos ángulos de obturador

seq_parameter_set_rbsp() {	Descriptor
<code>sps_max_sub_layers_minus1</code>	<code>u(3)</code>
<code>sps_reserved_zero_5bits</code>	<code>u(5)</code>
<code>profile_tier_level(sps_max_sub_layers_minus1)</code>	
<code>sps_seq_parameter_set_id</code>	<code>uc(v)</code>
...	
<code>sps_hfr_BC_enabled_flag</code>	<code>u(1)</code>
<code>if(sps_hfr_BC_enabled_flag) {</code>	<code>u(1)</code>
<code> sps_base_frame_rate</code>	<code>u(9)</code>
<code>}</code>	
...	
<code>sps_extension_flag</code>	<code>u(1)</code>
<code>if(sps_extension_flag)</code>	
<code> while(more_rbsp_data())</code>	
<code> sps_extension_data_flag</code>	<code>u(1)</code>
<code> rbsp_trailing_bits()</code>	
<code>}</code>	

Tabla 9: Sintaxis de RBSB de conjunto 17 parámetros de imagen de ejemplo para la entrada en diversos ángulos de obturador

<code>pic_parameter_set_rbsp() {</code>	Descriptor
<code>pps_pic_parameter_set_id</code>	<code>ue(v)</code>
<code>pps_seq_parameter_set_id</code>	<code>ue(v)</code>
<code>...</code>	
<code>if(sps_hfr_BC_enabled_flag)</code>	
<code>number_avg_frames</code>	<code>se(v)</code>
<code>...</code>	
<code>rbp_trailing_bits()</code>	
<code>}</code>	

Variaciones en la segunda realización (velocidad de fotograma fija)

- 5 El estándar de codificación HEVC (H.265) (ref. [1]) y el estándar de codificación de vídeo versátil en desarrollo (comúnmente denominada VVC, véase ref. [2]), definen un elemento sintáctico, **pic_struct**, que indica si una imagen debe mostrarse como un fotograma o como uno o más campos, y si una imagen decodificada debe repetirse. Se proporciona una copia de la Tabla D.2, "Interpretación de pic_struct," de HEVC para facilitar la referencia en el Apéndice.
- 10 Es importante observar que, tal como se aprecia por los inventores, el elemento sintáctico pic_struct existente puede soportar solo un subconjunto específico de velocidades de fotogramas de contenido cuando se usa un contenedor de codificación de velocidad de fotograma fija. Por ejemplo, cuando se usa un contenedor de velocidad de fotograma fija de 60 fps, la sintaxis pic_struct existente, cuando fixed_pic_rate_within_cvs_flag es igual a 1, puede soportar 30 fps usando duplicación de fotograma, y 24 fps usando duplicación de fotograma y triplicación de fotograma en combinación
- 15 alternante en cada fotograma adicional. Sin embargo, cuando se usa un contenedor de velocidad de fotograma fija de 120 fps, la sintaxis pic_struct actual no puede soportar velocidades de fotograma de 24 fps ni de 30 fps. Para aliviar este problema, se proponen dos nuevos métodos: uno es una ampliación de la versión de HEVC, y el otro no lo es.

Método 1: pic_struct sin compatibilidad con versiones anteriores

- 20 El VVC todavía está en desarrollo, por lo que se puede diseñar sintaxis con máxima libertad. En una realización, en pic_struct, se propone eliminar las opciones para duplicar fotogramas y triplicar fotogramas, usar un valor específico de pic_struct para indicar repetición de fotograma arbitraria y añadir un nuevo elemento sintáctico, **num_frame_repetition_minus2**, que especifica el número de fotogramas a repetir. Un ejemplo de la sintaxis propuesta se describe en las siguientes Tablas, donde la Tabla 10 denota cambios sobre la Tabla D.2.3 en HEVC y la Tabla 11 denota cambios de la Tabla D.2 mostrada en el Apéndice.

25

Tabla 10: Sintaxis de mensaje de SEI de temporización de imagen de ejemplo, método 1

pic_timing(payloadSize) {	Descriptor
if(frame_field_info_present_flag) {	
pic_struct	u(4)
<i>if(pic_struct == 7)</i>	<i>u(4)</i>
num_frame_repetition_minus2	<i>u(4)</i>
source_scan_type	u(2)
duplicate_flag	u(1)
}	
.... (as the original)	

num_frame_repetition_minus2 más 2 indica que cuando fixed_pic_rate_within_cvs_flag es igual a 1, el fotograma debe mostrarse num_frame_repetition_minus2 más 2 veces de manera consecutiva en visualizaciones con un intervalo de refresco de fotograma igual a DpbOutputElementalInterval[n] tal como viene dado por la ecuación E-73.

5

Tabla 11: Ejemplo de la revisión de pic_struct de acuerdo con el método 1

Valor	Visualización indicada de la imagen	Restricciones
0	fotograma (progresivo)	field_seq_flag será igual a 0
1	Campo superior	field_seq_flag será igual a 1
2	Campo inferior	field_seq_flag será igual a 1
3	Campo superior, campo inferior, en ese orden	field_seq_flag será igual a 0
4	Campo inferior, campo superior, en ese orden	field_seq_flag será igual a 0
5	Campo superior, campo inferior, campo superior repetido, en ese orden	field_seq_flag será igual a 0
6	Campo inferior, campo superior, campo inferior repetido, en ese orden	field_seq_flag será igual a 0
7	<i>Repetición de fotograma</i>	<i>field_seq_flag será igual a 0</i> <i>fixed_pic_rate_within_cvs_flag será igual a 1</i>
8	Campo superior emparejado con campo inferior anterior en el orden de salida	field_seq_flag será igual a 1
9	Campo inferior emparejado con campo superior anterior en orden de salida	field_seq_flag será igual a 1
10	Campo superior emparejado con siguiente campo inferior en el orden de salida	field_seq_flag será igual a 1

Valor	Visualización indicada de la imagen	Restricciones
11	Campo inferior emparejado con el siguiente campo superior en el orden de salida	field_seq_flag será igual a 1

Método 2: Versión ampliada de la versión de HEVC de pic_struct

Los decodificadores AVC y HEVC ya están implementados, por lo que puede desearse simplemente extender la sintaxis pic_struct existente sin eliminar opciones antiguas. En una realización, se añaden una nueva pic_struct = 13, un valor de "extensión de repetición de fotograma" y un nuevo elemento sintáctico, **num_frame_repetition_minus4**. Un ejemplo de la sintaxis propuesta se describe en las Tablas 12 y 13. Para los valores pic_struct 0-12, la sintaxis propuesta es idéntica a la de la Tabla D.2 (tal como se muestra en el Apéndice), por lo que esos valores se omiten por sencillez.

Tabla 12: sintaxis de mensaje de SEI de temporización de imagen de ejemplo, método 2

pic_timing(payloadSize) {	Descriptor
if(frame_field_info_present_flag) {	
pic_struct	u(4)
if(pic_struct == 13)	u(4)
num_frame_repetition_minus4	u(4)
source_scan_type	u(2)
duplicate_flag	u(1)
}	
... (as the original)	

num_frame_repetition_minus4 más 4 indica que cuando fixed_pic_rate_within_cvs_flag es igual a 1, el fotograma debe mostrarse (num_frame_repetition_minus4 más 4) veces de manera consecutiva en visualizaciones con un intervalo de refresco de fotograma igual a DpbOutputElementalInterval[n] tal como viene dado por la ecuación E-73.

Tabla 13: Ejemplo de método de pic_struct revisado, método 2

Valor	Visualización indicada de la imagen	Restricciones
0-12	Como en la Tabla D.2	Como en la Tabla D.2
13	Extensión de repetición de fotograma	field_seq_flag será igual a 0 fixed_pic_rate_within_cvs_flag será igual a 1

En HEVC, el parámetro frame_field_info_present_flag está presente en la información de usabilidad de vídeo (VUI), pero los elementos sintácticos pic_struct, source_scan_type y double_flag están en el mensaje de SEI pic_timing(). En una realización, se propone mover todos los elementos sintácticos relacionados a la VUI, junto con frame_field_info_present_flag. Un ejemplo de la sintaxis propuesta se representa en la Tabla 14.

Tabla 14: Sintaxis de parámetro de VUI de ejemplo con soporte para el elemento sintáctico *pic_struct* revisado

<code>vui_parameters() {</code>	Descriptor
<code>...</code>	<code>u(1)</code>
<code>field_seq_flag</code>	<code>u(1)</code>
<code>frame_field_info_present_flag</code>	<code>u(1)</code>
<code>if(frame_field_info_present_flag) {</code>	
<code> pic_struct</code>	<code>u(4)</code>
<code> source_scan_type</code>	<code>u(2)</code>
<code> duplicate_flag</code>	<code>u(1)</code>
<code>}</code>	
<code>...</code>	
<code>}</code>	

Señalización alternativa de la información del ángulo del obturador

- 5 Cuando se trata con velocidad de fotograma variable, es deseable identificar tanto la velocidad de fotograma deseada como el ángulo de obturador deseado. En las normas anteriores de codificación de vídeo, la "Información de Usabilidad de Vídeo" (VUI) proporciona información esencial para la visualización adecuada del contenido de vídeo, tal como la relación de aspecto, primarios de color, submuestreo de croma, etc. VUI también puede proporcionar información de velocidad de fotograma si la velocidad de pic fija se establece en 1; sin embargo, no hay soporte para la información de ángulo de obturador. Las realizaciones permiten que se usen diferentes ángulos de obturador para diferentes capas temporales, y un decodificador puede usar información de ángulo de obturador para mejorar el aspecto final en la pantalla.

15 Por ejemplo, HEVC admite subcapas temporales que usan esencialmente técnicas de descarte de fotogramas para pasar de una velocidad de fotograma más alta a una velocidad de fotograma más baja. El problema principal con esto es que el ángulo de obturador efectivo se reduce con cada descarte de fotograma. Como ejemplo, se pueden deducir 60 fps de un vídeo de 120 fps descartando cada fotograma adicional; se pueden deducir 30 fps descartando 3 de 4 fotogramas; y se pueden deducir 24 fps descartando 4 de 5 fotogramas. Suponiendo un obturador completo de 360 grados para 120 Hz, con descarte de fotograma simple, los ángulos de obturador para 60 fps, 30 fps y 24 fps son 180, 90 y 72 grados, respectivamente [3]. La experiencia ha demostrado que los ángulos de obturador inferiores a 180

20 grados son en general inaceptables, especialmente con velocidades de fotograma inferiores a 50 Hz. Proporcionando información del ángulo de obturador, por ejemplo, si se desea que una pantalla produzca un efecto cinematográfico a partir de un vídeo de 120 Hz con ángulo de obturador reducido para cada capa temporal, se pueden aplicar técnicas inteligentes para mejorar el aspecto final.

25 En otro ejemplo, se puede desear soportar una capa temporal diferente (digamos, un subflujo de bits de 60 fps dentro de un flujo de bits de 120 fps) con el mismo ángulo de obturador. Entonces, el problema principal es que cuando se visualiza vídeo de 120 fps a 120 Hz, los fotogramas pares/impares tienen un ángulo de obturador efectivo diferente. Si una pantalla tiene la información relacionada, se pueden aplicar técnicas inteligentes para mejorar el aspecto final. Un ejemplo de la sintaxis propuesta se muestra en la Tabla 15, donde la Tabla de sintaxis de parámetros de VUI E.2.1 en HEVC (Ref. [1]) se modifica para soportar información del ángulo del obturador tal como se ha indicado. Obsérvese

30 que en otra realización, en lugar de expresar la sintaxis *shutter_angle* en grados absolutos, puede expresarse alternativamente como la relación de la velocidad de fotograma con respecto a la velocidad de obturador (véase la ecuación (1)).

Tabla 15: Sintaxis de parámetro de VUI de ejemplo con soporte de ángulo de obturador

vui_parameters() {	Descriptor
...	
vui_timing_info_present_flag	u(1)
if(vui_timing_info_present_flag) {	
vui_num_units_in_tick	u(32)
vui_time_scale	u(32)
vui_poc_proportional_to_timing_flag	u(1)
if(vui_poc_proportional_to_timing_flag)	
vui_num_ticks_poc_diff_one_minus1	ue(v)
vui_hrd_parameters_present_flag	u(1)
if(vui_hrd_parameters_present_flag)	
hrd_parameters(1, sps_max_sub_layers_minus1)	
}	
vui_shutter_angle_info_present_flag	u(1)
if(vui_shutter_angle_info_present_flag) {	
fixed_shutter_angle_within_cvs_flag	u(1)
if(fixed_shutter_angle_within_cvs_flag)	
fixed_shutter_angle	u(9)
else {	
for(i = 0; i <= sps_max_sub_layers_minus1; i++) {	
sub_layer_shutter_angle[i]	u(9)
}	
}	
...	
}	

vui_shutter_angle_info_present_flag igual a 1 especifica que la información de ángulo de obturador está presente en la estructura sintáctica vui_parameters(). vui_shutter_angle_info_present_flag igual a 0 especifica que la información de ángulo de obturador no está presente en la estructura sintáctica vui_parameters().

fixed_shutter_angle_within_cvs_flag igual a 1 especifica que la información de ángulo de obturador es la misma para todas las subcapas temporales en la CVS. **fixed_shutter_angle_within_cvs_flag** igual a 0 especifica que la información de ángulo de obturador puede no ser la misma para todas las subcapas temporales en la CVS.

fixed_shutter_angle especifica el ángulo de obturador en grados dentro de una CVS. El valor de "fixed_shutter_angle" estará en el intervalo de 0 a 360.

sub_layer_shutter_angle[i] especifica el ángulo de obturador en grados cuando HighestTid es igual a i. El valor de sub_layer_shutter_angle[i] estará en el intervalo de 0 a 360.

Actualización gradual de la velocidad de fotograma dentro de una secuencia de vídeo codificada (CVS)

Los experimentos han mostrado que para contenido de HDR mostrado en una pantalla de HDR, para percibir el mismo temblor por movimiento que la reproducción de rango dinámico estándar (SDR) en una pantalla de 100 nits, la velocidad de fotograma debe aumentarse basándose en el brillo del contenido. En la mayoría de los estándares (AVC, HEVC, VVC, etc.), la velocidad de fotograma de vídeo puede indicarse en la VUI (contenida en el SPS) usando los elementos sintácticos **vui_time_scale**, **vui_num_units_in_tick** y **elemental_duration_in_tc_minus1[temporal_id_max]**, por ejemplo, tal como se muestra en la Tabla 16 a continuación (véase la Sección E.2.1 en la Ref. [1]).

Tabla 16: Elementos sintácticos de VUI para indicar la velocidad de fotograma en HEVC

vui_parameters() {	Descriptor
...	
vui_timing_info_present_flag	u(1)
if(vui_timing_info_present_flag) {	
vui_num_units_in_tick	u(32)
vui_time_scale	u(32)
vui_poc_proportional_to_timing_flag	u(1)
if(vui_poc_proportional_to_timing_flag)	
vui_num_ticks_poc_diff_one_minus1	ue(v)
vui_hrd_parameters_present_flag	u(1)
if(vui_hrd_parameters_present_flag)	
hrd_parameters(1, sps_max_sub_layers_minus1)	
}	
....	

Tal como se explica en la Ref. [1], la variable ClockTick se deduce de la siguiente manera y se denomina un tictac de reloj:

$$\text{ClockTick} = \text{vui_num_units_in_tick} \div \text{vui_time_scale}$$

$$\text{picture_duration} = \text{ClockTick} * (\text{elemental_duration_in_tc_minus1}[i] + 1)$$

$$\text{frame_rate} = 1/\text{pic_duration}.$$

Sin embargo, la velocidad de fotograma solo se puede cambiar en instantes de tiempo específicos, por ejemplo, en HEVC, solo en fotogramas de intra punto de acceso aleatorio (Intra Random Access Point, IRAP) o al inicio de una nueva CVS. Para la reproducción de HDR, cuando hay un caso de fundido de entrada o fundido de salida, debido a que el brillo de una imagen está cambiando fotograma a fotograma, podría haber una necesidad de cambiar la velocidad de fotograma o la duración de la imagen para cada imagen. Para permitir refrescar la velocidad de fotograma o la duración de la imagen en cualquier instante de tiempo (incluso fotograma a fotograma), en una realización, se propone un nuevo mensaje de SEI para “velocidad de refresco gradual”, tal como se muestra en la Tabla 17.

Tabla 17: Sintaxis de ejemplo para soportar la velocidad de fotograma de refresco gradual en mensajería de SEI

<code>gradual_refresh_rate(payloadSize) {</code>	Descriptor
num_units_in_tick	u(32)
time_scale	u(32)
<code>}</code>	

La definición de la nueva sintaxis **num_units_in_tick** es igual que la de `vui_num_units_in_tick`, y la definición de **time_scale** es la misma que la de `vui_time_scale`.

num_units_in_tick es el número de unidades de tiempo de un reloj que opera a la frecuencia `time_scale` Hz que corresponde a un incremento (denominado un tictac de reloj) de un contador de tictacs de reloj. `num_units_in_tick` será mayor que 0. Un tictac de reloj, en unidades de segundos, es igual al cociente de `num_units_in_tick` dividido por `time_scale`. Por ejemplo, cuando la velocidad de imagen de una señal de vídeo es de 25 Hz, `time_scale` puede ser igual a 27.000.000 y `num_units_in_tick` puede ser igual a 1.080.000 y, en consecuencia, un tictac de reloj puede ser igual a 0,04 segundos.

time_scale es el número de unidades de tiempo que pasan en un segundo. Por ejemplo, un sistema de coordenadas temporales que mide el tiempo usando un reloj de 27 MHz tiene un `time_scale` de 27.000.000. El valor de `time_scale` será mayor que 0.

El tiempo de duración de imagen para la imagen que usa el mensaje de SEI `gradual_refresh_rate` se define como:

$$\text{picture_duration} = \text{num_units_in_tick} \div \text{time_scale}$$

Señalización de información de ángulo de obturador a través de mensajería de SEI

Tal como se ha explicado anteriormente, la Tabla 15 proporciona un ejemplo de sintaxis de parámetros de VUI con soporte de ángulo de obturador. Como ejemplo, y sin limitación, la Tabla 18 enumera elementos sintácticos idénticos, pero ahora como parte de un mensaje de SEI para información de ángulo de obturador. Obsérvese que la mensajería de SEI se está usando solo como un ejemplo, y una mensajería similar puede construirse en otras capas de sintaxis de alto nivel, tal como el conjunto de parámetros de secuencia (SPS), el conjunto de parámetros de imagen (PPS), la cabecera de segmento o grupo de mosaicos y similares.

Tabla 18: Sintaxis de mensaje de SEI de ejemplo para información de ángulo de obturador

<code>shutter_angle_information(payloadSize) {</code>	Descriptor
fixed_shutter_angle_within_cvs_flag	u(1)
<code>if (fixed_shutter_angle_within_cvs_flag)</code>	
fixed_shutter_angle	u(9)
<code>else {</code>	
<code>for(i = 0; i <= sps_max_sub_layers_minus1; i++)</code>	
sub_layer_shutter_angle[i]	u(9)
<code>}</code>	
<code>}</code>	

El ángulo de obturador se expresa típicamente en grados de 0 a 360 grados. Por ejemplo, un ángulo de obturador de 180 grados indica que la duración de la exposición es la mitad de la duración del fotograma. El ángulo de obturador puede expresarse como: $\text{shutter_angle} = \text{frame_rate} * 360 * \text{shutter_speed}$, donde shutter_speed es la duración de la exposición y frame_rate es la inversa de la duración del fotograma. frame_rate para la subcapa temporal Tid dada puede indicarse mediante **num_units_in_tick**, **time_scale**, **elemental_duration_in_tc_minus1**[Tid]. Por ejemplo, cuando $\text{fixed_pic_rate_within_cvs_flag}[Tid]$ es igual a 1:

$$\text{frame_rate} = \text{time_scale} / (\text{num_units_in_tick} * (\text{elemental_duration_in_tc_minus1}[Tid] + 1)) .$$

En algunas realizaciones, el valor del ángulo de obturador (por ejemplo, $\text{fixed_shutter_angle}$) puede no ser un número entero, por ejemplo, puede ser de 135,75 grados. Para permitir más precisión, en la Tabla 21, se puede sustituir $u(9)$ (9 bits sin signo) por $u(16)$ o alguna otra profundidad de bits adecuada (por ejemplo, 12 bits, 14 bits o más de 16 bits).

En algunas realizaciones, puede ser beneficioso expresar la información del ángulo de obturador en términos de "tictacs de reloj". En VVC, la variable ClockTick se deduce de la siguiente manera:

$$\text{ClockTick} = \text{num_units_in_tick} \div \text{time_scale} \quad . \quad (8)$$

Entonces, se puede expresar tanto la duración del fotograma como la duración de la exposición como múltiplo o fraccionario de tictacs de reloj:

$$\text{exposure_duration} = fN * \text{ClockTick} \quad , \quad (9)$$

$$\text{frame_duration} = fM * \text{ClockTick} \quad . \quad (10)$$

donde fN y fM son valores de coma flotante y $fN \leq fM$.

Entonces

$$\begin{aligned} \text{shutter_angle} &= \text{frame_rate} * 360 * \text{shutter_speed} = \\ &= (1/\text{frame_duration}) * 360 * \text{exposure_duration} = \\ &= (\text{exposure_duration} * 360) / \text{frame_duration} = \\ &= (fN * \text{ClockTick} * 360) / (fM * \text{ClockTick}) = \\ &= (fN/fM) * 360 = (\text{Numerator}/\text{Denominator}) * 360, \end{aligned} \quad (11)$$

donde numerador y denominador son números enteros que se aproximan a la relación fN/fM .

La Tabla 19 muestra un ejemplo de mensajería de SEI indicada por la ecuación (11). En este ejemplo, el ángulo de obturador debe ser mayor que 0 para una cámara del mundo real.

Tabla 19: Ejemplo de mensajería de SEI para información de ángulo de obturador basada en tictacs de reloj

shutter_angle_information (payloadSize) {	Descriptor
fixed_shutter_angle_within_cvs_flag	u(1)
if (fixed_shutter_angle_within_cvs_flag) {	
fixed_shutter_angle_numer_minus1	u(16)
fixed_shutter_angle_denom_minus1	u(16)
}	
else {	
for(i = 0; i <= sps_max_sub_layers_minus1; i++) {	
sub_layer_shutter_angle_numer_minus1[i]	u(16)
sub_layer_shutter_angle_denom_minus1[i]	u(16)
}	
}	

Tal como se ha descrito anteriormente, el uso de u(16) (16 bits sin signo) para la precisión del ángulo del obturador se representa como un ejemplo y corresponde a una precisión de: $360/2^{16} = 0,0055$. La precisión se puede ajustar basándose en aplicaciones reales. Por ejemplo, usando u(8), la precisión es de $360/2^8 = 1.4063$.

NOTA - El ángulo de obturador se expresa en grados mayores que 0 pero menores o iguales que 360 grados. Por ejemplo, un ángulo de obturador de 180 grados indica que la duración de la exposición es la mitad de la duración del fotograma.

fixed_shutter_angle_within_cvs_flag Igual a 1 especifica que el valor del ángulo de obturador es el mismo para todas las subcapas temporales en la CVS. **fixed_shutter_angle_within_cvs_flag** igual a 0 especifica que el valor del ángulo de obturador puede no ser el mismo para todas las subcapas temporales en la CVS.

fixed_shutter_angle_numer_minus1 más 1 especifica el numerador usado para deducir el valor del ángulo de obturador. El valor de **fixed_shutter_angle_numer_minus1** estará en el intervalo de 0 a 65535, ambos inclusive.

fixed_shutter_angle_denom_minus1 más 1 especifica el denominador usado para deducir el valor del ángulo del obturador. El valor de **fixed_shutter_angle_denom_minus1** estará en el intervalo de 0 a 65535, ambos inclusive.

El valor de **fixed_shutter_angle_numer_minus1** será menor o igual que el valor de **fixed_shutter_angle_denom_minus1**.

La variable shutter_angle en grados se deduce como sigue:

$$\text{shutterAngle} = 360 * (\text{fixed_shutter_angle_numer_minus1} + 1) \div (\text{fixed_shutter_angle_denom_minus1} + 1)$$

sub_layer_shutter_angle_numer_minus1[i] más 1 especifica el numerador usado para deducir el valor del ángulo obturador cuando HighestTid es igual a i. El valor de

sub_layer_shutter_angle_numer_minus1[i] estará en el intervalo de 0 a 65535, inclusive.

sub_layer_shutter_angle_denom_minus1[i] más 1 especifica el denominador usado para deducir el valor del ángulo de obturador cuando HighestTid es igual a i. El valor de

sub_layer_shutter_angle_denom_minus1[i] estará en el intervalo de 0 a 65535, inclusive.

El valor de **sub_layer_shutter_angle_numer_minus1[i]** será menor o igual que el valor de **sub_layer_shutter_angle_denom_minus1[i]**.

La variable subLayerShutterAngle[i] en grados se deduce como sigue:

$$\text{subLayerShutterAngle}[i] = 360 * (\text{sub_layer_shutter_angle_numer_minus1}[i] + 1) \div (\text{sub_layer_shutter_angle_denom_minus1}[i] + 1)$$

En otra realización, la duración de fotograma (por ejemplo, frame_duration) puede especificarse por algún otro medio. Por ejemplo, en DVB/ATSC, cuando fixed_pic_rate_within_cvs_flag [Tid] es igual a 1:

$$\text{frame_rate} = \text{time_scale} / (\text{num_units_in_tick} * (\text{elemental_duration_in_tc_minus1}[\text{Tid}] + 1)),$$

$$\text{frame_duration} = 1 / \text{frame_rate}.$$

La sintaxis en la Tabla 19 y en algunas de las Tablas posteriores supone que el ángulo de obturador siempre será mayor que cero; sin embargo, el ángulo de obturador = 0 puede usarse para señalar una intención creativa, donde el contenido debe mostrarse sin ningún desenfoque por movimiento. Tal podría ser el caso para gráficos en movimiento, animación, texturas de CGI y pantallas mate, etc. De este modo, por ejemplo, señalar ángulo de obturador = 0 podría ser útil para la decisión de modo en un transcodificador (por ejemplo, para seleccionar modos de transcodificación que conserven bordes) así como en una pantalla que reciba los metadatos de ángulo de obturador a través de una interfaz CTA o una interfaz 3GPP. Por ejemplo, el ángulo de obturador = 0 podría usarse para indicar a una pantalla que no debe realizar ningún procesamiento de movimiento tal como eliminación de ruido, interpolación de fotograma, y similares. En tal realización, los elementos sintácticos **fixed_shutter_angle_numer_minus1** y **sub_layer_shutter_angle_numer_minus1[i]** pueden ser reemplazados por los elementos sintácticos **fixed_shutter_angle_numer** y **sub_layer_shutter_angle_numer[i]**, en donde **fixed_shutter_angle_numer** especifica el numerador utilizado para deducir el valor del ángulo de obturador. El valor de fixed_shutter_angle_numer estará en el intervalo de 0 a 65535, ambos inclusive.

sub_layer_shutter_angle_numer[i] especifica el numerador usado para deducir el valor del ángulo obturador cuando HighestTid es igual a i. El valor de sub_layer_shutter_angle_numer[i] estará en el intervalo de 0 a 65535, ambos inclusive.

En otra realización, fixed_shutter_angle_denom_minus1 y sub_layer_shutter_angle_denom_minus1[i] también pueden ser reemplazados por los elementos sintácticos fixed_shutter_angle_denom y sub_layer_shutter_angle_denom[i].

En una realización, tal como se representa en la Tabla 20, se pueden reutilizar las sintaxis num_units_in_tick y time_scale definidas en SPS ajustando general_hrd_parameters_present_flag igual a 1 en VVC. En este escenario, el mensaje de SEI puede volver a nombrarse como mensaje de SEI de duración de exposición.

Tabla 20: Ejemplo de mensajería de SEI para la duración de la exposición de la señalización

exposure_duration_information (payloadSize) {	Descriptor
fixed_exposure_duration_within_cvs_flag	u(1)
if (fixed_shutter_angle_within_cvs_flag) {	
fixed_exposure_duration_numer_minus1	u(16)
fixed_exposure_duration_denom_minus1	u(16)
}	
else {	
for(i = 0; i <= sps_max_sub_layers_minus1; i++) {	
sub_layer_exposure_duration_numer_minus1[i]	u(16)
sub_layer_exposure_duration_denom_minus1[i]	u(16)
}	
}	

fixed_exposure_duration_within_cvs_flag igual a 1 especifica que el valor de duración de la exposición efectivo es el mismo para todas las subcapas temporales en la CVS. **fixed_exposure_duration_within_cvs_flag** igual a 0 especifica que el valor de duración de la exposición efectivo puede no ser el mismo para todas las subcapas temporales en la CVS.

fixed_exposure_duration_numer_minus1 más 1 especifica el numerador usado para deducir el valor de duración de la exposición. El valor de **fixed_exposure_duration_numer_minus1** estará en el intervalo de 0 a 65535, ambos inclusive.

fixed_exposure_duration_denom_minus1 más 1 especifica el denominador usado para deducir el valor de duración de la exposición. El valor de **fixed_exposure_duration_denom_minus1** estará en el intervalo de 0 a 65535, ambos inclusive.

El valor de **fixed_exposure_during_numer_minus1** será menor o igual que el valor de **fixed_exposure_duration_denom_minus1**.

La variable **fixedExposureDuration** se deduce de la siguiente manera:

$$\text{fixedExposureDuration} = (\text{fixed_exposure_duration_numer_minus1} + 1 \\) \div (\text{fixed_exposure_duration_denom_minus1} + 1) * \text{ClockTicks}$$

sub_layer_exposure_duration_numer_minus1[i] más 1 especifica el numerador usado para deducir el valor de duración de la exposición cuando **HighestTid** es igual a *i*. El valor de **sub_layer_exposure_duration_numer_minus1[i]** estará en el intervalo de 0 a 65535, ambos inclusive.

sub_layer_exposure_duration_denom_minus1[i] más 1 especifica el denominador usado para deducir el valor de duración de la exposición cuando **HighestTid** es igual a *i*. El valor de **sub_layer_exposure_duration_denom_minus1[i]** estará en el intervalo de 0 a 65535, ambos inclusive.

El valor de **sub_layer_exposure_duration_numer_minus1[i]** será menor o igual que el valor de **sub_layer_exposure_duration_denom_minus1[i]**.

La variable **subLayerExposureDuration[i]** para **HighestTid** igual a *i* se deduce como sigue:

$$\text{subLayerExposureDuration}[i] = (\text{sub_layer_exposure_duration_numer_minus1}[i] + 1 \\) \div (\text{sub_layer_exposure_duration_denom_minus1}[i] + 1) * \text{ClockTicks}.$$

En otra realización, tal como se muestra en la Tabla 21, se puede definir explícitamente clockTick por los elementos sintácticos expo_num_units_in_tick y expo_time_scale. La ventaja en este caso es que no depende de si general_hrd_parameters_present_flag se establece igual a 1 en VVC, como en la realización anterior, entonces

$$\text{clockTick} = \text{expo_num_units_in_tick} \div \text{expo_time_scale} \quad . \quad (12)$$

5 **Tabla 21: Ejemplo de mensajería de SEI para señalización de tiempo de exposición**

exposure_duration_information (payloadSize) {	Descriptor
expo_num_units_in_tick	u(32)
expo_time_scale	u(32)
fixed_exposure_duration_within_cvs_flag	u(1)
if (!fixed_exposure_duration_within_cvs_flag)	
for(i = 0; i <= sps_max_sub_layers_minus1; i++) {	
sub_layer_exposure_duration_numer_minus1[i]	u(16)
sub_layer_exposure_duration_denom_minus1[i]	u(16)
}	
}	

expo_num_units_in_tick es el número de unidades de tiempo de un reloj que opera en la frecuencia time_scale Hz que corresponde a un incremento (llamado un tictac de reloj) de un contador de tictacs de reloj. expo_num_units_in_tick será mayor que 0. Un tictac de reloj, definido por la variable clockTick, en unidades de segundos, es igual al cociente de expo_num_units_in_tick dividido por expo_time_scale.

expo_time_escala es el número de unidades de tiempo que pasan en un segundo.

$$\text{clockTick} = \text{expo_num_units_in_tick} \div \text{expo_time_scale}.$$

NOTA: Los dos elementos sintácticos: expo_num_units_in_tick y expo_time_scale se definen para medir la duración de exposición.

Es un requisito para la conformidad del flujo de bits que clockTick sea menor o igual que ClockTick cuando num_units_in_tick y time_scale estén presentes.

fixed_exposure_duration_within_cvs_flag igual a 1 especifica que el valor de la duración de exposición efectivo es el mismo para todas las subcapas temporales en la CVS.

fixed_exposure_duration_within_cvs_flag igual a 0 especifica que el valor de la duración de exposición efectivo puede no ser el mismo para todas las subcapas temporales en la CVS. Cuando fixed_exposure_duration_within_cvs_flag es igual a 1, la variable fixedExposureDuration se establece igual a clockTick.

sub_layer_exposure_duration_numer_minus1[i] más 1 especifica el numerador usado para deducir el valor de la duración de exposición cuando HighestTid es igual a i. El valor de sub_layer_exposure_duration_numer_minus1[i] estará en el rango de 0 a 65535, ambos inclusive.

sub_layer_exposure_duration_denom_minus1[i] más 1 especifica el denominador usado para deducir el valor de la duración de exposición cuando HighestTid es igual a i. El valor de sub_layer_exposure_duration_denom_minus1[i] estará en el rango de 0 a 65535, ambos inclusive.

El valor de sub_layer_exposure_duration_numer_minus1 [i] será menor o igual que el valor de sub_layer_exposure_duration_denom_minus1[i].

La variable `subLayerExposureDuration[i]` para `HighestTid` igual a `i` se describe como sigue:

$$\text{subLayerExposureDuration}[i] = (\text{sub_layer_exposure_duration_numer_minus1}[i] + 1) \div (\text{sub_layer_exposure_duration_denom_minus1}[i] + 1) * \text{clockTick}.$$

Tal como se ha explicado anteriormente, los parámetros sintácticos

5 `sub_layer_exposure_duration_numer_minus1[i]` y `sub_layer_exposure_duration_denom_minus1[i]` también puede reemplazarse por `sub_layer_exposure_duration_numer[i]` y `sub_layer_exposure_duration_denom[i]`.

En otra realización, tal como se muestra en la Tabla 22, se puede definir el parámetro `ShutterInterval` (es decir, duración de la exposición) por los elementos sintácticos `sii_num_units_in_shutter_interval` y `sii_time_scale` donde $\text{ShutterInterval} = \text{sii_num_units_in_shutter_interval} \div \text{sii_time_scale}$. (13)

10 **Tabla 22: Ejemplo de mensajería de SEI para la señalización de la duración de exposición (información de intervalo de obturador)**

<code>shutter_interval_information (payloadSize) {</code>	Descriptor
<code>sii_num_units_in_shutter_interval</code>	<code>u(32)</code>
<code>sii_time_scale</code>	<code>u(32)</code>
<code>fixed_shutter_interval_within_cvs_flag</code>	<code>u(1)</code>
<code>if (!fixed_shutter_interval_within_cvs_flag)</code>	
<code>for(i = 0; i <= sps_max_sub_layers_minus1; i++) {</code>	
<code>sub_layer_shutter_interval_numer[i]</code>	<code>u(16)</code>
<code>sub_layer_shutter_interval_denom[i]</code>	<code>u(16)</code>
<code>}</code>	

Semántica de mensaje de SEI de información de intervalo de obturador

15 El mensaje de SEI de información de intervalo de obturador indica el intervalo de obturador para el contenido de vídeo asociado antes de codificar y visualizar, por ejemplo, para contenido capturado por cámara, la cantidad de tiempo que un sensor de imagen estuvo expuesto para producir una imagen.

20 **`sii_num_units_in_shutter_interval`** especifica el número de unidades de tiempo de un reloj que opera a la frecuencia `sii_time_scale` Hz que corresponde a un incremento de un contador de ticks de reloj de obturador. El intervalo de obturador, definido por la variable `ShutterInterval`, en unidades de segundos, es igual al cociente de `sii_num_units_in_shutter_interval` dividido por `sii_time_scale`. Por ejemplo, cuando `ShutterInterval` es igual a 0,04 segundos, `sii_time_scale` puede ser igual a 27.000.000 y `sii_num_units_in_shutter_interval` puede ser igual a 1.080.000.

`sii_time_scale` especifica el número de unidades de tiempo que pasan en un segundo. Por ejemplo, un sistema de coordenadas de tiempo que mide el tiempo usando un reloj de 27 MHz tiene un `sii_time_scale` de 27.000.000. Cuando el valor de `sii_time_scale` es mayor que 0, el valor de `ShutterInterval` se especifica mediante:

25
$$\text{ShutterInterval} = \text{sii_num_units_in_shutter_interval} \div \text{sii_time_scale}$$

De lo contrario (el valor de `sii_time_scale` es igual a 0), `ShutterInterval` debe interpretarse como desconocido o no especificado.

NOTA 1 - Un valor de `ShutterInterval` igual a 0 puede indicar que el contenido de vídeo asociado contiene contenido de captura de pantalla, contenido generado por ordenador u otro contenido de no captura de cámara.

30 NOTA 2 - Un valor de `ShutterInterval` mayor que el valor de la inversa de la velocidad de imagen codificada, el intervalo de imagen codificada, puede indicar que la velocidad de imagen codificada es mayor que la velocidad de imagen a la que se creó el contenido de vídeo asociado - por ejemplo, cuando la velocidad de imagen codificada es de 120 Hz y la velocidad de imagen del contenido de vídeo asociado antes de la codificación y visualización es de 60 Hz. El

intervalo codificado para la subcapa temporal dada Tid puede indicarse mediante ClockTick y elemental_duration_in_tc_minus1 [Tid]. Por ejemplo, cuando fixed_pic_rate_within_cvs_flag [Tid] es igual a 1, el intervalo de imagen para la subcapa temporal Tid dada, definido por la variable PictureInterval [Tid], puede especificarse mediante:

5
$$\text{PictureInterval}[Tid] = \text{ClockTick} * (\text{elemental_duration_in_tc_minus1}[Tid] + 1) .$$

fixed_shutter_interval_within_cvs_flag igual a 1 especifica que el valor de ShutterInterval es el mismo para todas las subcapas temporales en la CVS. fixed_shutter_interval_within_cvs_flag igual a 0 especifica que el valor de ShutterInterval puede no ser el mismo para todas las subcapas temporales en la CVS.

10 **sub_layer_shutter_interval_numer[i]** especifica el numerador utilizado para deducir el intervalo de obturador de la subcapa, definido por la variable subLayerShutterInterval[i], en unidades de segundos, cuando HighestTid es igual a i.

sub_layer_shutter_interval_denom[i] especifica el denominador utilizado para deducir el intervalo de obturador de la subcapa, definido por la variable subLayerShutterInterval[i], en unidades de segundos, cuando HighestTid es igual a i.

15 El valor de subLayerShutter[i] para HighestTid igual a i se deduce como sigue. Cuando el valor de fixed_shutter_interval_within_cvs_flag es igual a 0 y el valor de sub_layer_shutter_interval_denom[i] es mayor que 0:

$$\text{subLayerShutterInterval}[i] = \text{ShutterInterval} * \text{sub_layer_shutter_interval_numer}[i] \\ \div \text{sub_layer_shutter_interval_denom}[i]$$

20 De lo contrario (el valor de sub_layer_shutter_interval_denom[i] es igual a 0), subLayerShutterInterval[i] debe interpretarse como no reconocida o no especificada. Cuando el valor de fixed_shutter_interval_within_cvs_flag no es igual a 0,

$$\text{subLayerShutterInterval}[i] = \text{ShutterInterval}.$$

En una realización alternativa, en lugar de usar un numerador y un denominador para señalar el intervalo de obturador de la subcapa, se usa un solo valor. Un ejemplo de tal sintaxis se muestra en la Tabla 23.

Tabla 23: Ejemplo de mensajería de SEI para señalización de intervalo de obturador

shutter_interval_information (payloadSize) {	Descriptor
sii_num_units_in_shutter_interval	u(32)
sii_time_scale	u(32)
fixed_shutter_interval_within_cvs_flag	u(1)
if (!fixed_shutter_interval_within_cvs_flag)	
for(i = 0; i <= sps_max_sub_layers_minus1; i++) {	
sub_layer_num_units_in_shutter_interval[i]	u(32)
}	
}	

Semántica de mensaje de SEI de información de intervalo de obturador

El mensaje de SEI de información de intervalo de obturador indica el intervalo de obturador para el contenido de vídeo asociado antes de codificar y visualizar, por ejemplo, para contenido capturado por cámara, la cantidad de tiempo que un sensor de imagen estuvo expuesto para producir una imagen.

30 **sii_num_units_in_shutter** especifica el número de unidades de tiempo de un reloj que opera a la frecuencia sii_time_scale Hz que corresponde a un incremento de un contador de tictacs de reloj de obturador. El intervalo de obturador, definido por la variable ShutterInterval, en unidades de segundos, es igual al cociente de sii_num_units_in_shutter_interval dividido por sii_time_scale. Por ejemplo, cuando ShutterInterval es igual a 0,04

segundos, `sii_time_scale` puede ser igual a 27.000.000 y `sii_num_units_in_shutter_interval` puede ser igual a 1.080.000.

`sii_time_scale` especifica el número de unidades de tiempo que pasan en un segundo. Por ejemplo, un sistema de coordenadas de tiempo que mide el tiempo usando un reloj de 27 MHz tiene un `sii_time_scale` de 27.000.000. Cuando el valor de `sii_time_scale` es mayor que 0, el valor de `ShutterInterval` se especifica mediante:

$$\text{ShutterInterval} = \text{sii_num_units_in_shutter_interval} \div \text{sii_time_scale}$$

De lo contrario (el valor de `sii_time_scale` es igual a 0), `ShutterInterval` debe interpretarse como desconocido o no especificado.

NOTA 1 - Un valor de `ShutterInterval` igual a 0 puede indicar que el contenido de vídeo asociado contiene contenido de captura de pantalla, contenido generado por ordenador u otro contenido de no captura de cámara.

NOTA 2 - Un valor de `ShutterInterval` mayor que el valor de la inversa de la velocidad de imagen codificada, el intervalo de imagen codificada, puede indicar que la velocidad de imagen codificada es mayor que la velocidad de imagen a la que se creó el contenido de vídeo asociado - por ejemplo, cuando la velocidad de imagen codificada es 120 Hz y la velocidad de imagen del contenido de vídeo asociado antes de la codificación y visualización es 60 Hz. El intervalo de imagen codificada para la subcapa temporal `Tid` dada puede indicarse mediante `ClockTick` y `elemental_duration_in_tc_minus1 [Tid]`. Por ejemplo, cuando `fixed_pic_rate_within_cvs_flag [Tid]` es igual a 1, el intervalo de imagen para la subcapa temporal `Tid` dada, definido por la variable `PictureInterval[Tid]`, puede especificarse por:

$$\text{PictureInterval}[Tid] = \text{ClockTick} * (\text{elemental_duration_in_tc_minus1}[Tid] + 1)$$

`fixed_shutter_interval_within_cvs_flag` Igual a 1 especifica que el valor de `ShutterInterval` es el mismo para todas las subcapas temporales en la CVS. `fixed_shutter_interval_within_cvs_flag` igual a 0 especifica que el valor de `ShutterInterval` puede no ser el mismo para todas las subcapas temporales en la CVS.

`sub_layer_num_units_in_shutter_interval[i]` especifica el número de unidades de tiempo de un reloj que opera a la frecuencia `sii_time_scale` Hz que corresponde a un incremento de un contador de tictacs de reloj de obturador. El intervalo de obturador de subcapa, definido por la variable `subLayerShutterInterval[i]`, en unidades de segundos, cuando `HighestTid` es igual a `i`, es igual al cociente de `sub_layer_num_units_in_shutter_interval[i]` dividido por `sii_time_scale`.

Cuando el valor de `fixed_shutter_interval_within_cvs_flag` es igual a 0 y el valor de `sii_time_scale` es mayor que 0, el valor de `subLayersShutterInterval[i]` se especifica mediante:

$$\text{subLayerShutterInterval}[i] = \text{sub_layer_num_units_in_shutter_interval}[i] \div \text{sii_time_scale}$$

De lo contrario (el valor de `sii_time_scale` es igual a 0), `subLayersShutterInterval[i]` debe interpretarse como desconocido o no especificado. Cuando el valor de `fixed_shutter_interval_within_cvs_flag` no es igual a 0,

$$\text{subLayerShutterInterval}[i] = \text{ShutterInterval}.$$

La Tabla 24 proporciona un resumen de los seis enfoques explicados en las Tablas 18-23 para proporcionar mensajería de SEI relacionada con el ángulo de obturador o la duración de la exposición.

Tabla 24: Resumen de enfoques de mensajería de SEI para señalar información de ángulo de obturador de señal

Tabla No.	Elementos de señalización clave y dependencias
18	El ángulo de obturador (0 a 360) se señala explícitamente
19	El ángulo de obturador se expresa como una relación de valores de numerador y denominador que van a escalarse en 360 (está implícito el valor de tictac de reloj)
20	La duración de la exposición se señala como una relación de valores de numerador y denominador (está implícito el valor de tictac de reloj)

21	La duración de la exposición se señala como una relación de valores de numerador y denominador; el valor de tictac de reloj se señala explícitamente como una relación de dos valores
22	La información del intervalo de obturador se señala como la relación de dos valores: el número de unidades de tictac de reloj en la exposición y una escala de tiempo de exposición; los tiempos de exposición relacionados con la subcapa se señalizan como una relación de dos valores
23	La información del intervalo de obturador o la duración de la exposición se señala como la relación de dos valores: el número de unidades de tictac de reloj en la exposición y una escala de tiempo de exposición; los tiempos de exposición relacionados con la subcapa se señalizan como el número de unidades de tictac de reloj en la exposición en cada subcapa

Señalización de velocidad de fotograma variable

Tal como se explica en el documento solicitud provisional de U.S. 62/883.195, presentada el 16 de agosto de 2019, en muchas aplicaciones, se desea que un decodificador soporte la reproducción a velocidades de fotograma variables. La adaptación de la velocidad de fotograma es típicamente parte de las operaciones en el decodificador de referencia (HRD) hipotético, como se describe, por ejemplo, en el Anexo C de la Ref. [2]. En una realización, se propone señalar a través de mensajería de SEI u otros medios un elemento sintáctico que define el tiempo de presentación de imagen (Picture Presentation Time, PPT) como función de un reloj de 90 kHz. Este es el tipo de repetición del tiempo de salida de la memoria intermedia de imágenes del decodificador (DPB) nominal como se especifica en el HRD, pero ahora usando una precisión de ClockTicks de 90 kHz como se especifica en el sistema MPEG-2. El beneficio de este mensaje de SEI es a) si la HRD no está habilitada, todavía se puede usar el mensaje de SEI de PPT para indicar la temporización para cada fotograma; b) puede facilitar la traducción de la temporización del flujo de bits y la temporización del sistema.

La Tabla 25 describe un ejemplo de la sintaxis del mensaje de temporización de PPT propuesto, que coincide con la sintaxis de la variable marca de tiempo de presentación (1presentation Time Stamp, PTS) que se usa en el transporte MPEG-2 (H.222) (Ref. [4]).

Tabla 25: Sintaxis de ejemplo para mensajería de tiempo de presentación de imagen

picture_presentation_time (payloadSize) {	Descriptor
PPT	u(33)
}	

PPT (tiempo de presentación de imagen)

Los tiempos de presentación estarán relacionados con los tiempos de decodificación de la siguiente manera: El PPT es un número de 33 bits codificado en tres campos separados. Indica el momento de la presentación, $tp_n(k)$, en el decodificador objetivo de sistema de una unidad de presentación k de flujo elemental n . El valor de PPT se especifica en unidades del período de la frecuencia de reloj del sistema dividido por 300 (rendimiento 90 kHz). El tiempo de presentación de imagen se deduce del PPT de acuerdo con la ecuación siguiente.

$$PPT(k) = ((system_clock_frequency \times tp_n(k))/300) \% 2^{33}$$

donde $tp_n(k)$ es el tiempo de presentación de la unidad de presentación $P_n(k)$.

Mensajería de intervalo de obturador en AVC

En una realización, si existe un mensaje de SEI de información de intervalo de obturador (Shutter Interval Information, SII) para cualquier imagen en una secuencia de vídeo codificada (CVS), entonces se sugiere que debe existir en la primera unidad de acceso de la CVS. A diferencia de la HEVC, un índice temporal (que se usa para identificar un índice de subcapa) no existe en un flujo de bits de una sola capa de AVC. Para abordar este problema, cuando el intervalo de obturador no está fijo dentro de una CVS, se propone que un mensaje de SEI de información de intervalo de obturador esté presente para cada imagen para asignar un valor para `sii_sub_layer_idx` a cada imagen para identificar el índice de subcapa de la imagen actual. Otra información relacionada con el intervalo de obturador se presentará solo para la primera unidad de acceso de la CVS, y persistirá hasta que comience una nueva CVS o finalice el flujo de bits.

En AVC, una unidad de acceso se define como un conjunto de unidades de NAL que son consecutivas en orden de decodificación y contienen exactamente una imagen codificada primaria. Además de la imagen codificada primaria, una unidad de acceso también puede contener una o más imágenes codificadas redundantes, una imagen codificada auxiliar u otras unidades de NAL que no contienen fragmentos o particiones de datos de fragmentos de una imagen codificada. La decodificación de una unidad de acceso siempre da como resultado una imagen decodificada.

Los valores de elementos sintácticos de ejemplo para el caso en el que el intervalo de obturador se fija para la CVS se muestran en la Tabla 26. En la Tabla 27 se muestran valores de elemento sintáctico de ejemplo para el primer y posterior mensaje de SEI de información de intervalo de obturador para el caso en el que el intervalo de obturador pueda ser diferente para diferentes subcapas. En las Tablas 26 y 27, las celdas con "(ninguno)" indican que no se señala ningún valor en el mensaje de SEI de información de intervalo de obturador para el elemento sintáctico correspondiente.

Tabla 26: Ejemplo de valores de elemento sintáctico de mensaje de SEI de información de intervalo de obturador para la unidad de acceso IDR de intervalo de obturador fijo

elemento de sintaxis	1er mensaje de SEI de información de intervalo de obturador en la CVS
sii_sub_layer_idx	0
shutter_interval_info_present_flag	1
sii_time_scale	u(32)
fixed_shutter_interval_within_cvs_flag	1
sii_num_units_in_shutter_interval	u(32)
sii_max_sub_layers_minus1	(ninguno)
sub_layer_num_units_in_shutter_interval[i]	(ninguno)

Tabla 27: Ejemplo de valores de elementos sintácticos de mensaje de SEI de información de intervalo de obturador para intervalo de obturador no fijo para unidad de acceso IDR

elemento de sintaxis	1er mensaje de SEI de información de intervalo de obturador en la CVS	Mensajes SEI de información de intervalo de obturador posteriores en la CVS	
sii_sub_layer_idx	0	0	ue(v) > 0
shutter_interval_info_present_flag	1	0	(ninguno)
sii_time_scale	u(32)	(ninguno)	(ninguno)
fixed_shutter_interval_within_cvs_flag	0	(ninguno)	(ninguno)
sii_num_units_in_shutter_interval	(ninguno)	(ninguno)	(ninguno)
sii_max_sub_layers_minus1	u (3)	(ninguno)	(ninguno)
sub_layer_num_units_in_shutter_interval[i]	u(32)	(ninguno)	(ninguno)

La Tabla 28 representa la estructura sintáctica de la invención para la mensajería de SEI de SII en AVC.

Tabla 28: sintaxis de mensaje de SEI de SII en AVC

shutter_interval_info(payloadSize) {	C	Descriptor
sii_sub_layer_idx	5	ue(v)
if(sii_sub_layer_idx == 0)		
shutter_interval_info_present_flag	5	u(1)
if(shutter_interval_info_present_flag)		
sii_time_scale	5	u(32)
fixed_shutter_interval_within_cvs_flag	5	u(1)
if(fixed_shutter_interval_within_cvs_flag)		
sii_num_units_in_shutter_interval	5	u(32)
else {		
sii_max_sub_layers_minus1	5	u(3)
for(i = 0; i <= sii_max_sub_layers_minus1; i++)		
sub_layer_num_units_in_shutter_interval[i]	5	u(32)
}		
}		
}		
}		

El mensaje de SEI de información de intervalo de obturador indica el intervalo de obturador para las imágenes de origen de vídeo asociadas antes de codificar y visualizar, por ejemplo, para contenido capturado por cámara, el intervalo de obturador es la cantidad de tiempo que un sensor de imagen está expuesto para producir cada imagen de origen.

sii_sub_layer_idx especifica el índice de subcapa temporal de intervalo de obturador de la imagen actual. El valor de **sii_sub_layer_idx** será igual a 0 cuando la unidad de acceso actual sea la primera unidad de acceso de la CVS. Cuando **fixed_shutter_interval_within_cvs_flag** es igual a 1, el valor de **sii_sub_layer_idx** será igual a 0. De lo contrario, **fixed_shutter_interval_within_cvs_flag** es igual a 0, el valor de **sii_sub_layer_idx** será menor o igual que el valor de **sii_max_sub_layers_minus1**.

shutter_interval_info_present_flag igual a 1 indica que los elementos sintácticos **sii_time_scale**, **fixed_shutter_interval_within_cvs_flag** y **sii_num_units_in_shutter_interval** o **sii_max_sub_layers_minus1** y **sub_layer_num_units_in_shutter_interval[i]** están presentes. **shutter_interval_info_present_flag** igual a 0 indica que los elementos sintácticos **sii_time_scale**, **fixed_shutter_interval_within_cvs_flag**, **sii_num_units_in_shutter_interval**, **sii_max_sub_layers_minus1** y **sub_layer_num_units_in_shutter_interval[i]** no están presentes. El valor de **shutter_interval_info_present_flag** será igual a 1 cuando la unidad de acceso actual sea la primera unidad de acceso de la CVS. De lo contrario, la unidad de acceso actual no es la primera unidad de acceso de la CVS, el valor de **shutter_interval_info_present_flag** será igual a 0.

sii_time_scale especifica el número de unidades de tiempo que pasan en un segundo. El valor de **sii_time_scale** será mayor que 0. Por ejemplo, un sistema de coordenadas de tiempo que mide el tiempo usando un reloj de 27 MHz tiene una **sii_time_scale** de 27.000.000.

5 **fixed_shutter_interval_within_cvs_flag** Igual a 1 especifica que el intervalo de obturador indicado es el mismo para todas las imágenes en la CVS. **fixed_shutter_interval_within_cvs_flag** igual a 0 especifica que el intervalo de obturador indicado puede no ser el mismo para todas las imágenes en la CVS.

10 **sii_num_units_in_shutter_interval**, cuando **fixed_shutter_interval_within_cvs_flag** es igual a 1, especifica el número de unidades de tiempo de un reloj que opera a la frecuencia **sii_time_scale** Hz que corresponde al intervalo de obturador indicado de cada imagen en la CVS. El valor 0 puede usarse para indicar que el contenido de vídeo asociado contiene contenido de captura de pantalla, contenido generado por ordenador u otro contenido no capturado por cámara.

15 El intervalo de obturador indicado, indicado por la variable **ShutterInterval**, en unidades de segundos, es igual al cociente de **sii_num_units_in_shutter_interval** dividido por **sii_time_scale**. Por ejemplo, para representar un intervalo de obturador igual a 0,04 segundos, **sii_time_scale** puede ser igual a 27.000.000 y **sii_num_units_in_shutter_interval** puede ser igual a 1.080.000.

sii_max_sub_layers_minus1 más 1 especifica el número máximo de índices de subcapas temporales de intervalo de obturador que pueden estar presentes en la CVS.

20 **sub_layer_num_units_in_shutter_interval[i]**, cuando está presente, especifica el número de unidades de tiempo de un reloj que opera a la frecuencia **sii_time_scale** Hz que corresponde al intervalo de obturador de cada imagen en la CVS para el que el valor de **sii_sub_layer_idx** es igual a *i*. El intervalo de obturador de subcapa para cada imagen para la que el valor de **sii_sub_layer_idx** es igual a *i*, indicado por la variable **subLayerShutterInterval[i]**, en unidades de segundos, es igual al cociente de **sub_layer_num_units_in_shutter_interval[i]** dividido por **sii_time_scale**.

La variable **subLayerShutterInterval[i]**, correspondiente al intervalo de obturador indicado de cada imagen en la representación de subcapa con **TemporalId** igual a *i* en la CVS, se deduce por lo tanto de la siguiente manera:

```

if( fixed_shutter_interval_within_cvs_flag )
    subLayerShutterInterval[ i ] = sii_num_units_in_shutter_interval ÷
    sii_time_scale
else
    subLayerShutterInterval[ i ] = sub_layer_num_units_in_shutter_interval[ i ] ÷
    sii_time_scale

```

25 Cuando un mensaje de SEI de información de intervalo de obturador está presente para cualquier unidad de acceso en una CVS, un mensaje de SEI de información de intervalo de obturador estará presente para la unidad de acceso de IDR que es la primera unidad de acceso de la CVS. Todos los mensajes de SEI de información de intervalo de obturador que se aplican a la misma unidad de acceso tendrán el mismo contenido.

30 **sii_time_scale** y **fixed_shutter_interval_within_cvs_flag** persisten desde la primera unidad de acceso de la CVS hasta que comienza una nueva CVS o termina el flujo de bits.

35 Cuando el valor de **fixed_shutter_interval_within_cvs_flag** es igual a 0, un mensaje de SEI de información de intervalo de obturador estará presente para cada imagen en la CVS. Cuando está presente, **sii_num_units_in_shutter_interval**, **sii_max_sub_layers_minus1** y **sub_layer_num_units_in_shutter_interval[i]**, persisten desde la primera unidad de acceso de la CVS hasta que comienza una nueva CVS o finaliza el flujo de bits.

Referencias

- [1] *High efficiency video coding*, H.265, Serie H, Codificación de vídeo en movimiento, ITU, (02/2018).
- [2] B. Bross, J. Chen, y S. Liu, “*Versatile Video Coding (Draft 5)*”, documento emitido por JVET, JVET-N1001, v5, subido el 14 de mayo, 2019.
- 40 [3] C. Carbonara, J. DeFilippis, M. Korpi, “*High Frame Rate Capture and Production*”, SMPTE 2015 Annual Technical Conference and Exhibition, 26-29 de octubre, 2015.
- [4] *Infraestructure of Audiovisual Services – Transmission multiplexing and synchronization*, H.222.0, Serie H, Codificación genérica de imagen en movimiento e información de audio asociada: Sistemas, ITU, 08/2018.

IMPLEMENTACIÓN DE SISTEMA INFORMÁTICO EJEMPLAR

Las realizaciones de la presente invención pueden implementarse con un sistema informático, sistemas configurados en circuitos y componentes electrónicos, un dispositivo de circuito integrado (IC) tal como un microcontrolador, una matriz de puertas programables in situ (Field Programmable Gate Array, FPGA) u otro dispositivo lógico configurable o programable (Programmable Logic Device, PLD), un procesador de señales digitales o de tiempo discreto (Digital Signal Processor, DSP), un IC específico de aplicación (Application Specific Integrated Circuit, ASIC) y/o aparato que incluya uno o más de tales sistemas, dispositivos o componentes. El ordenador y/o IC pueden realizar, controlar o ejecutar instrucciones relacionadas con la escalabilidad de velocidad de fotograma, tales como las descritas en el presente documento. El ordenador y/o IC pueden calcular cualquiera de una variedad de parámetros o valores que se refieren a la escalabilidad de velocidad de fotograma descrita en el presente documento. Las realizaciones de imagen y vídeo pueden implementarse en hardware, software, firmware y diversas combinaciones de los mismos.

Ciertas implementaciones de la invención comprenden procesadores informáticos que ejecutan instrucciones de software que hacen que los procesadores realicen un método de la invención. Por ejemplo, uno o más procesadores en una pantalla, un codificador, un decodificador, un transcodificador o similares pueden implementar métodos relacionados con la escalabilidad de velocidad de fotograma como se describió anteriormente ejecutando instrucciones de software en una memoria de programa accesible para los procesadores. Las realizaciones de la invención también pueden proporcionarse en forma de un producto de programa. El producto de programa puede comprender cualquier medio no transitorio y tangible que lleve un conjunto de señales legibles por ordenador que comprendan instrucciones que, cuando se ejecuten por un procesador de datos, hagan que el procesador de datos ejecute un método de la invención. Los productos de programa de acuerdo con la invención pueden estar en cualquiera de una amplia variedad de formas no transitorias y tangibles. El producto de programa puede comprender, por ejemplo, medios físicos tales como medios de almacenamiento de datos magnéticos que incluyen disquetes, unidades de disco duro, medios de almacenamiento de datos ópticos que incluyen CD ROM, DVD, medios de almacenamiento de datos electrónicos que incluyen ROM, RAM flash o similares. Las señales legibles por ordenador en el producto de programa pueden opcionalmente comprimirse o cifrarse.

Cuando un componente (por ejemplo, un módulo de software, procesador, conjunto, dispositivo, circuito, etc.) se ha mencionado anteriormente, a menos que se indique lo contrario, la referencia a ese componente (incluyendo una referencia a un "medio") debe interpretarse como que incluye como equivalentes de ese componente cualquier componente que realice la función del componente descrito (por ejemplo, que sea funcionalmente equivalente), incluyendo componentes que no sean estructuralmente equivalentes a la estructura divulgada que realiza la función en las realizaciones de ejemplo ilustradas de la invención.

EQUIVALENTES, EXTENSIONES, ALTERNATIVAS Y MISCELÁNEOS

Se describen por lo tanto realizaciones de ejemplo que se refieren a la escalabilidad de velocidad de fotograma. En la memoria descriptiva anterior, se han descrito realizaciones de la presente invención con referencia a numerosos detalles específicos que pueden variar de implementación a implementación. Por lo tanto, el único indicador exclusivo de lo que es la invención y de lo que los solicitantes pretenden que sea la invención es el conjunto de reivindicaciones que se emiten a partir de esta solicitud, en la forma específica en la que se emiten tales reivindicaciones, incluyendo cualquier corrección posterior. Cualquier definición expuesta expresamente en el presente documento para los términos contenidos en dichas reivindicaciones gobernará el significado de dichos términos tal como se usan en las reivindicaciones. Por lo tanto, ninguna limitación, elemento, propiedad, característica, ventaja o atributo que no se mencione expresamente en una reivindicación debe limitar el alcance de tal reivindicación en modo alguno. La memoria descriptiva y los dibujos, por consiguiente, deben considerarse en un sentido ilustrativo y no restrictivo.

Apéndice

Este Apéndice proporciona una copia de la Tabla D.2 e información relacionada con `pic_struct` asociada de la especificación H.265 (Ref. [1]).

Tabla D.2 - Interpretación de `pic_Struct`

Valor	Visualización indicada de la imagen	Restricciones
0	(progresivo) fotograma	<code>field_seq_flag</code> será igual a 0
1	Campo superior	<code>field_seq_flag</code> será igual a 1
2	Campo inferior	<code>field_seq_flag</code> será igual a 1
3	Campo superior, campo inferior, en ese orden	<code>field_seq_flag</code> será igual a 0
4	Campo inferior, campo superior, en ese orden	<code>field_seq_flag</code> será igual a 0
5	Campo superior, campo inferior, campo superior repetido, en ese orden	<code>field_seq_flag</code> será igual a 0
6	Campo inferior, campo superior, campo inferior repetido, en ese orden	<code>field_seq_flag</code> será igual a 0
7	Duplicación de fotograma	<code>field_seq_flag</code> será igual a 0 <code>fixed_pic_rate_within_cvs_flag</code> será igual a 1
8	Triplicación de fotograma	<code>field_seq_flag</code> será igual a 0 <code>fixed_pic_rate_within_cvs_flag</code> será igual a 1
9	Campo superior emparejado con el campo inferior anterior en el orden de salida	<code>field_seq_flag</code> será igual a 1
10	Campo inferior emparejado con el campo superior anterior en orden de salida	<code>field_seq_flag</code> será igual a 1
11	Campo superior emparejado con siguiente campo inferior en el orden de salida	<code>field_seq_flag</code> será igual a 1
12	Campo inferior emparejado con siguiente campo superior en el orden de salida	<code>field_seq_flag</code> será igual a 1

5

Semántica del elemento sintáctico `pic_struct`

`pic_struct` indica si una imagen debe mostrarse como una fotograma o como uno o más campos y, para la visualización de fotogramas cuando `fixed_pic_rate_within_cvs_flag` es igual a 1, puede indicar un período de repetición de duplicación o de triplicación de fotograma para visualizaciones que usan un intervalo de refresco de fotograma fijo igual a `DpbOutputElementalInterval[n]` tal como viene dado por la Ecuación E-73. La interpretación de `pic_struct` se especifica en la Tabla D.2. Los valores de `pic_struct` que no se enumeran en la Tabla D.2 se reservan para uso futuro por ITU-T | ISO/IEC y no estarán presentes en flujos de bits que se ajustan a esta versión de esta memoria descriptiva. Los decodificadores ignorarán los valores reservados de `pic_struct`.

10

Cuando está presente, es un requisito de conformidad del flujo de bits que el valor de `pic_struct` debe restringirse de tal manera que exactamente una de las siguientes condiciones es verdadera:

15

- El valor de `pic_struct` es igual a 0, 7 u 8 para todas las imágenes en la CVS.
- El valor de `pic_struct` es igual a 1, 2, 9, 10, 11 o 12 para todas las imágenes en la CVS.

- El valor de `pic_struct` es igual a 3, 4, 5 o 6 para todas las imágenes en la CVS.

Cuando la `fixed_pic_rate_within_cvs_flag` es igual a 1, la duplicación de fotograma se indica mediante `pic_struct` igual a 7, lo que indica que el fotograma debe mostrarse dos veces consecutivas en pantallas con un intervalo de refresco de fotograma igual a `DpbOutputElementalInterval[n]` tal como viene dado por la ecuación E-73, y la triplicación de fotograma se indica mediante `pic_struct` igual a 8, lo que indica que el fotograma debe mostrarse tres veces consecutivas en pantallas con un intervalo de refresco de fotograma igual a `DpbOutputElementalInterval[n]` tal como viene dado por la ecuación E-73.

NOTA 3 - Se puede utilizar la duplicación de fotograma para facilitar la visualización, por ejemplo, de vídeo de exploración progresiva de 25 Hz en una pantalla de exploración progresiva de 50 Hz o vídeo de exploración progresiva de 30 Hz en una pantalla de exploración progresiva de 60 Hz. El uso de duplicación de fotograma y triplicación de fotograma en combinación alterna en cada trama adicional puede usarse para facilitar la presentación de vídeo de exploración progresiva de 24 Hz en una presentación de exploración progresiva de 60 Hz.

Las ubicaciones de muestreo verticales y horizontales nominales de muestras en los campos superior e inferior para los formatos de croma 4:2:0, 4:2:2 y 4:4:4 se muestran en la figura D.1, la figura D.2 y la figura D.3, respectivamente.

Los indicadores de asociación para campos (`pic_struct` igual a 9 a 12) proporcionan indicios para asociar campos de paridad complementaria entre sí como fotogramas. La paridad de un campo puede ser superior o inferior, y la paridad de dos campos se considera complementaria cuando la paridad de un campo es superior y la paridad del otro campo es inferior.

Cuando `frame_field_info_present_flag` es igual a 1, es un requisito de conformidad del flujo de bits que se aplicarán las restricciones especificadas en la tercera columna de la Tabla D.2.

NOTA 4 - Cuando `frame_field_info_present_flag` es igual a 0, entonces en muchos casos los valores por defecto pueden inferirse o indicarse por otros medios. En ausencia de otras indicaciones del tipo de presentación visual previsto de una imagen, el decodificador debe deducir el valor de `pic_struct` igual a 0 cuando `frame_field_info_present_flag` es igual a 0.

REIVINDICACIONES

1. Un aparato para procesar un flujo de vídeo codificado usando un procesador, comprendiendo el aparato:

Una entrada configurada para recibir un flujo de bits codificado que comprende una sección de imagen codificada que incluye una secuencia de vídeo codificada de imágenes de vídeo codificadas usando codificación de vídeo avanzada, AVC, y una sección de señalización que comprende una sección de mensajería de información de mejora suplementaria, SEI, que incluye parámetros de intervalo de obturador, en donde los parámetros de intervalo de obturador comprenden:

un índice de subcapa temporal de intervalo de obturador de una imagen actual en el flujo de bits codificado, en el que

si el índice de subcapa temporal del intervalo de obturador es igual a cero, entonces los parámetros del intervalo de obturador incluyen un indicador de presencia de información del intervalo de obturador, en donde si el indicador de presencia de información del intervalo de obturador es igual a uno, entonces los parámetros del intervalo de obturador incluyen:

un parámetro de escala temporal de intervalo de obturador que indica el número de unidades de tiempo que pasan en un segundo;

un indicador de duración de intervalo de obturador fijo que indica si la información de duración de intervalo de obturador es fija para todas las imágenes en la secuencia de vídeo codificada; y

si el indicador de duración de intervalo de obturador fijo indica que la información de duración de intervalo de obturador es fija, entonces los parámetros de intervalo de obturador incluyen

un parámetro de tictacs de reloj de intervalo de obturador que indica un número de unidades de tiempo de un reloj que opera a la frecuencia del parámetro de escala de tiempo de intervalo de obturador, en el que el parámetro de tictacs de reloj de intervalo de obturador dividido por el parámetro de escala de tiempo de intervalo de obturador indica un valor de duración de la exposición para todas las imágenes de vídeo en la secuencia de vídeo codificada,

en caso contrario, los parámetros de intervalo de obturador incluyen un elemento sintáctico que especifica un número máximo menos 1 de índices de subcapa temporal de intervalo de obturador que pueden estar presentes en la secuencia de vídeo codificada, y los parámetros de intervalo de obturador incluyen una matriz de parámetros de tictac de reloj de intervalo de obturador de subcapa que indican un número de unidades de tiempo de un reloj a la frecuencia del parámetro de escala de tiempo de intervalo de obturador para cada subcapa i en la secuencia de vídeo codificada,

de i igual a 0 hasta el número máximo menos 1 de índices de subcapa temporal de intervalo de obturador, en donde, para una primera subcapa en la secuencia de vídeo codificada, un parámetro de tictac de reloj de intervalo de obturador de subcapa correspondiente dividido por el parámetro de escala de tiempo de intervalo de obturador indica el valor de duración de exposición para todas las imágenes de vídeo en la primera subcapa de la secuencia de vídeo codificada; y

un procesador, en donde el procesador está configurado para decodificar la secuencia de imágenes de vídeo basándose en los parámetros de intervalo de obturador.

2. El aparato de la reivindicación 1, en donde la secuencia de vídeo codificada comprende dos o más unidades de acceso, y el índice de subcapa temporal de intervalo de obturador es 0 para una primera unidad de acceso entre las dos o más unidades de acceso, y no cero en caso contrario.

3. El aparato de la reivindicación 1, en donde el índice de subcapa temporal de intervalo de obturador es 0 cuando el indicador de duración de intervalo de obturador fijo indica que la información de duración de intervalo de obturador es fija.

4. El aparato de la reivindicación 1, en donde si el indicador de duración de intervalo de obturador fijo es igual a 1, entonces el índice de subcapa temporal del intervalo de obturador es igual a cero, de otro modo, si el indicador de duración de intervalo de obturador fijo es igual a 0, entonces el valor del índice de subcapa temporal del intervalo de obturador es menor o igual al número máximo menos 1 de los índices de subcapa temporal del intervalo de obturador.

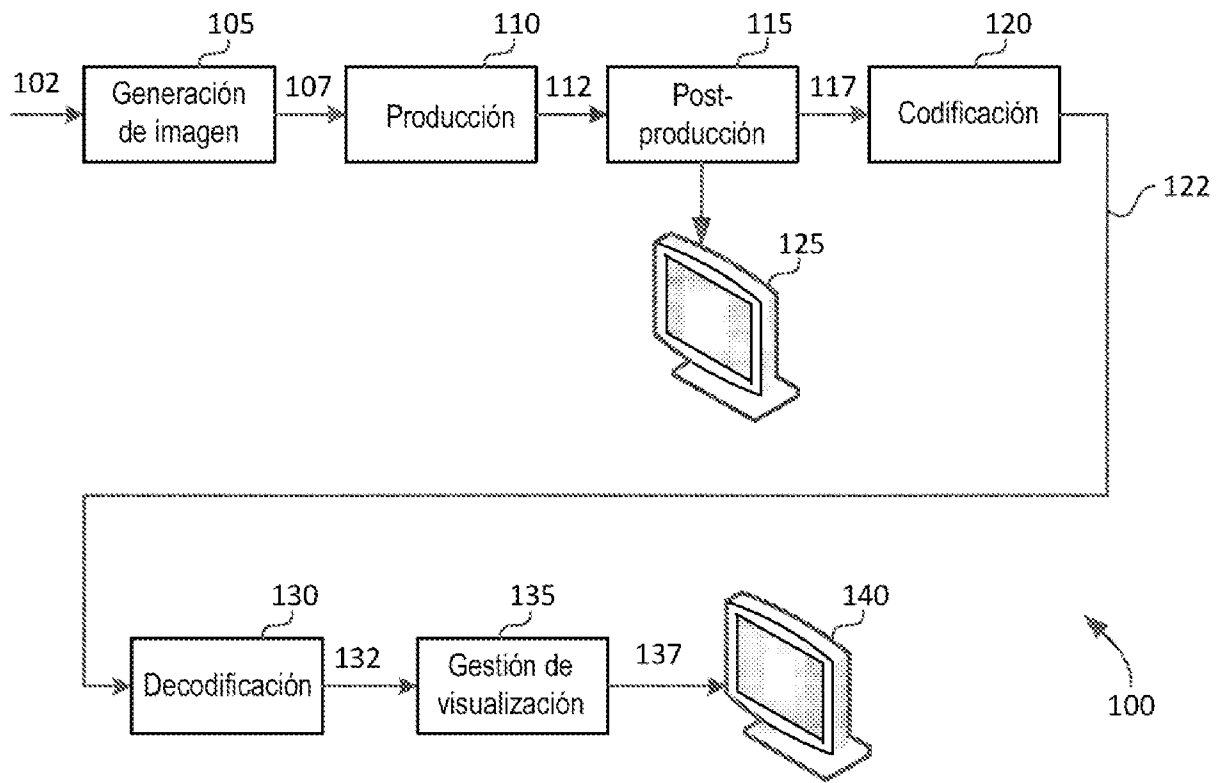


FIG. 1

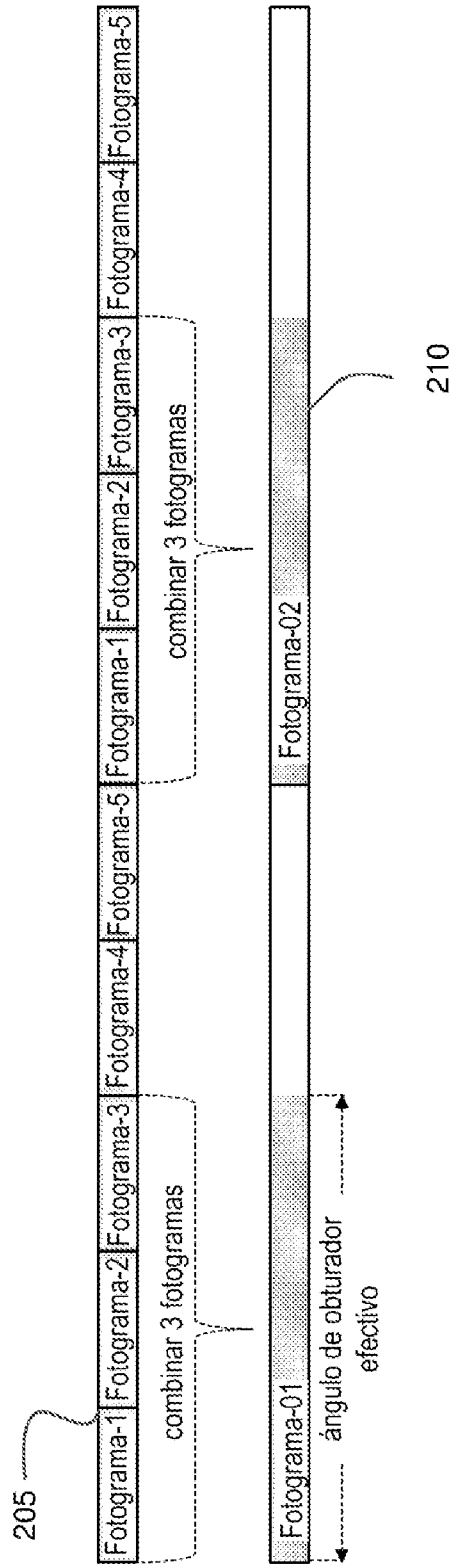


FIG. 2

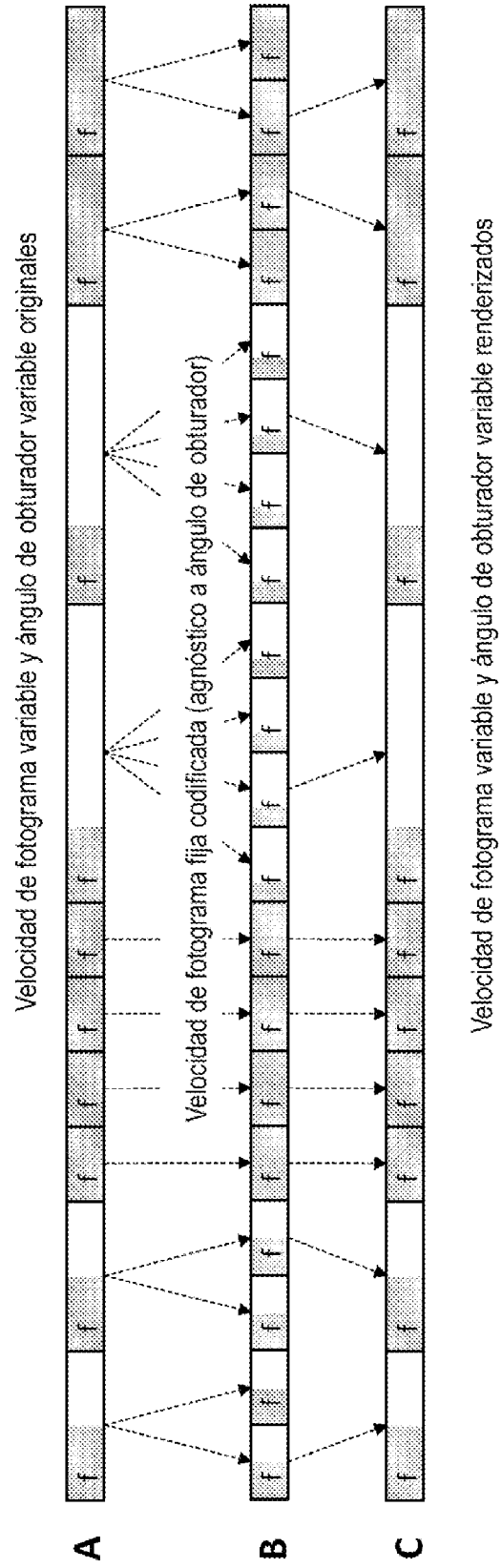


FIG. 3

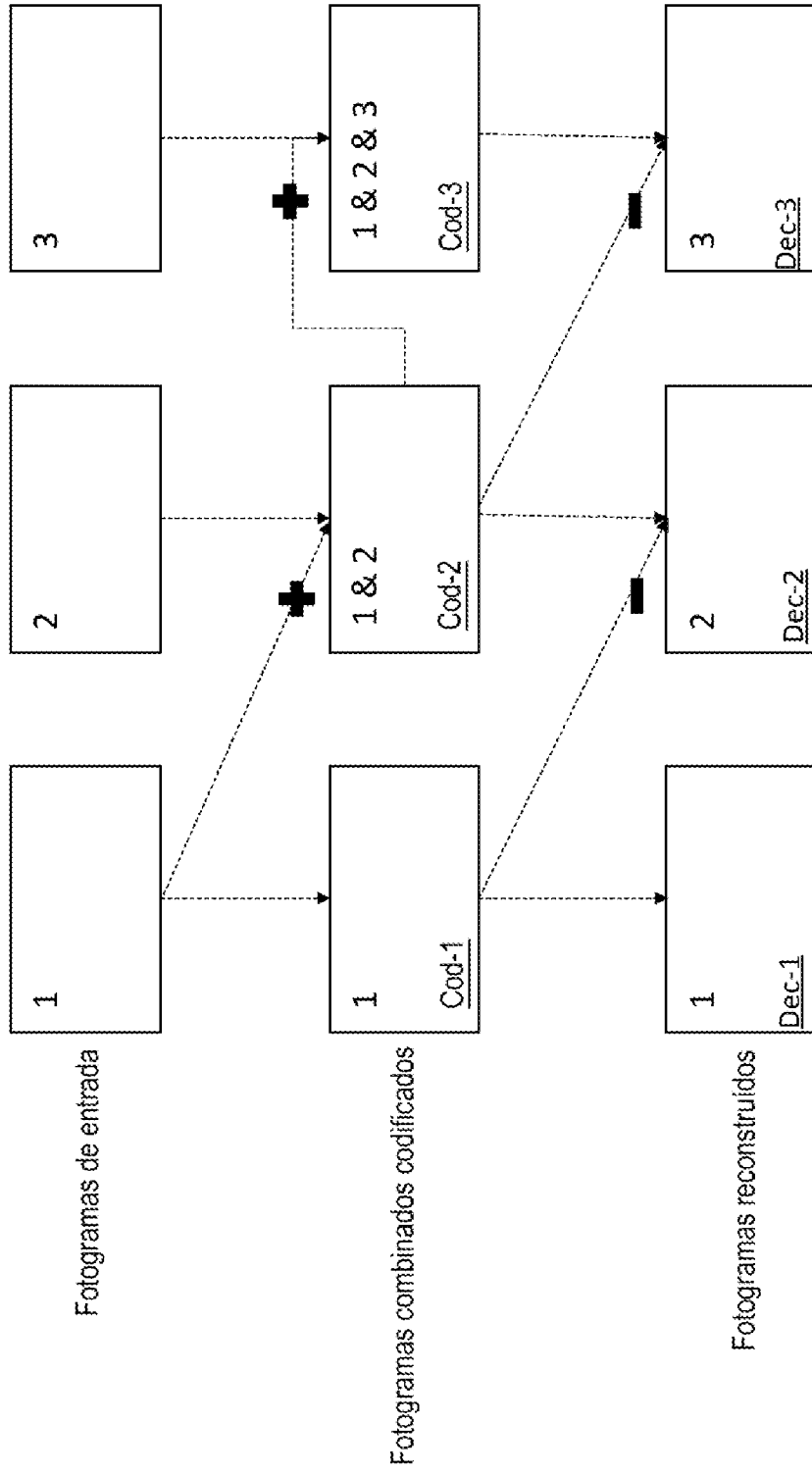


FIG. 4