

19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 637 166**

51 Int. Cl.:

**H04N 19/176** (2014.01)  
**H04N 19/70** (2014.01)  
**H04N 19/117** (2014.01)  
**H04N 19/463** (2014.01)  
**H04N 19/82** (2014.01)  
**H04N 19/42** (2014.01)

12

## TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **15.03.2013** **PCT/SE2013/050270**

87 Fecha y número de publicación internacional: **10.10.2013** **WO13151481**

96 Fecha de presentación y número de la solicitud europea: **15.03.2013** **E 13762601 (6)**

97 Fecha y número de publicación de la concesión europea: **10.05.2017** **EP 2834980**

54 Título: **Filtrado adaptativo de muestra con corrimientos**

30 Prioridad:

**05.04.2012 US 201261620639 P**  
**16.04.2012 US 201261624726 P**

45 Fecha de publicación y mención en BOPI de la  
traducción de la patente:  
**11.10.2017**

73 Titular/es:

**TELEFONAKTIEBOLAGET LM ERICSSON (PUBL)**  
**(100.0%)**  
**164 83 Stockholm, SE**

72 Inventor/es:

**ANDERSSON, KENNETH;**  
**SJÖBERG, RICKARD y**  
**WENNERSTEN, PER**

74 Agente/Representante:

**LINAGE GONZÁLEZ, Rafael**

ES 2 637 166 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

Filtrado adaptativo de muestra con corrimientos

## 5 Campo técnico

Las realizaciones del presente documento se refieren al procesamiento de datos de vídeo. En particular, se divulgan un método, un dispositivo de control de filtrado, un programa informático y un producto de programa informático para procesar una primera imagen en un procedimiento de filtrado adaptativo de muestra.

## 10 Antecedentes

Los datos de vídeo necesitan ser procesados en muchas situaciones y aplicaciones diferentes. Un tipo muy común de procesamiento de datos de vídeo es la codificación y decodificación de datos de vídeo, normalmente con el propósito de comprimir los datos de vídeo en el lado de origen/codificador mediante codificación de vídeo, y descomprimir los datos de vídeo codificados en el lado de destino/decodificador mediante decodificación de vídeo.

La codificación de vídeo de alta eficiencia (HEVC), también conocida como H.265, es un estándar de compresión de vídeo. La HEVC es desarrollada conjuntamente por el grupo de expertos en imágenes en movimiento (MPEG) de ISO/IEC y el grupo de expertos en codificación de vídeo (VCEG) del sector de normalización de las telecomunicaciones (ITU-T) como ISO/IEC 23008-2 MPEG-H Parte 2 e ITU-T H.265. MPEG y VCEG han establecido un equipo de colaboración conjunta en codificación de vídeo (JCT-VC) para desarrollar el estándar HEVC.

En un sistema de codificación o compresión de vídeo compatible con, por ejemplo, el estándar HEVC, los datos de vídeo se someten a diversos pasos de procesamiento, incluyendo por ejemplo predicción, residual, transformación, cuantificación, desbloqueo y filtrado adaptativo de bucles. A lo largo de la trayectoria de procesamiento en el sistema de codificación o compresión de vídeo, ciertas características de los datos de vídeo pueden ser alteradas a partir de los datos de vídeo originales debido a los accionamientos en los pasos de procesamiento al que se someten los datos de vídeo. Por ejemplo, pueden producirse artefactos en forma de cambios en la intensidad de la imagen (por ejemplo, crominancia o luminancia) para los píxeles en una trama de vídeo y/o entre tramas de vídeo sucesivas. Dichos artefactos pueden ser visualmente notorios; por lo tanto, se pueden tomar medidas para compensar los artefactos en un intento de eliminar o al menos aliviarlos.

En HEVC, se utiliza [2] un esquema de compensación de intensidad conocido como corrimiento (o desfase) adaptativo de muestras (SAO). El esquema SAO [1] clasifica cada píxel en los datos de vídeo en una de varias categorías SAO de acuerdo con un contexto dado. El contexto puede ser, por ejemplo, la intensidad de píxel de los datos de vídeo, que a menudo se denomina "corrimientos de banda SAO". Alternativamente o adicionalmente, el contexto puede ser una relación de valor de píxel entre el píxel actual y sus píxeles vecinos, que a menudo se denomina "corrimientos de borde SAO". En este último caso, las categorías SAO representan artefactos de borde típicos y están asociadas con respectivos valores de corrimiento correspondientes para ser aplicados a los píxeles en la respectiva categoría SAO para compensar el artefacto de borde en cuestión. Dependiendo de dónde se aplique el corrimiento adaptativo, los datos de vídeo pueden representar datos de vídeo reconstruidos, datos de vídeo que han sufrido un desbloqueo, datos de vídeo filtrados en bucle adaptativo u otros datos de vídeo en una etapa intermedia durante el proceso de codificación o decodificación. Por lo tanto, SAO se puede utilizar en HEVC después del proceso de filtro de desbloqueo si se utiliza un filtro de desbloqueo, de lo contrario directamente después de la reconstrucción de la predicción y residual. SAO modifica la imagen que se va a mostrar o almacenar en un búfer de imagen de referencia. El búfer de imagen de referencia mantiene las imágenes para su uso posterior para la inter-predicción de otras imágenes. Si se utiliza el filtrado adaptativo de bucle normalmente se realiza después de SAO.

Más específicamente, la compensación SAO en HEVC involucra cuatro categorías de corrimiento de borde SAO [1]-[3]. La primera categoría representa un caso en el que el píxel actual (o más específicamente su valor de intensidad) está en un mínimo local comparado con sus dos píxeles vecinos en una dirección seleccionada -horizontal (0 grados), vertical (90 grados) o diagonal (135 ó 45 grados)-. La segunda categoría representa un caso en el que el píxel actual es igual a uno de sus vecinos pero inferior al otro vecino en la dirección seleccionada. La tercera categoría representa un caso en el que el píxel actual es igual a uno de sus vecinos pero más alto que el otro vecino en la dirección seleccionada. La cuarta categoría representa un caso en el que el píxel actual está en un máximo local en comparación con sus dos píxeles vecinos en la dirección seleccionada.

Los filtros de desbloqueo se utilizan en los estándares de codificación de vídeo para combatir los artefactos de bloqueo. Los artefactos de bloqueo surgen porque las tramas de vídeo originales se dividen en bloques que se procesan de forma relativamente independiente. Los artefactos de bloqueo pueden, por ejemplo, surgir debido a diferentes intra-predicciones de los bloques, efectos de cuantificación y compensación de movimiento.

En HEVC hay un filtro de desbloqueo, también denominado filtro de bucle, después de la predicción y la reconstrucción residual, pero antes del almacenamiento de la reconstrucción para referencia posterior cuando se

codifican o decodifican las tramas subsiguientes. El filtrado de desbloqueo consta de varios pasos como decisiones de filtro, accionamientos de filtrado, una función de recorte y cambio de valores de píxeles.

Actualmente el corrimiento de banda y de borde SAO en HEVC tiene cuatro corrimientos por unidad mayor de codificación (LCU). La codificación de estos corrimientos es una sobrecarga que hace que muchas LCU eviten el uso de corrimientos de borde o de banda SAO. La codificación de varios corrimientos también consume memoria. Un problema es por lo tanto que varios corrimientos consumen demasiada memoria.

Otro problema es que cuesta muchos bits utilizar corrimiento de banda o de borde con grandes magnitudes para SAO en HEVC. Para velocidades de bits bajas donde el corrimiento con grandes magnitudes es más probable causa gastos innecesarios o menos uso de corrimientos con grandes magnitudes

Otro problema con el corrimiento de banda es que requiere una búsqueda de tabla basada en píxeles para realizar el accionamiento de corrimiento de banda. La tabla tiene un tamaño de 32 y necesita actualizarse antes de procesar cada LCU.

Otro problema adicional con el corrimiento de borde y banda es que requiere accionamientos de recorte para permitir que una modificación con corrimiento de banda permanezca dentro de la profundidad de bits de la señal de vídeo. Para el corrimiento de banda en el equipo lógico de referencia HEVC, esto se hace teniendo una tabla más grande  $32 + \text{maxoffset}/8 - \text{minoffset}/8$  donde los extremos de la tabla van al valor mínimo, respectivamente, máximo de acuerdo con la profundidad de bit. Para el corrimiento de borde, esto se hace teniendo una tabla tan grande como el intervalo de valores de píxeles  $+ \text{maxoffset} - \text{minoffset}$ . Esto no se menciona en el borrador de trabajo de HEVC.

## Sumario

Un objeto es mejorar un procedimiento SAO del tipo mencionado anteriormente con el fin de aliviar o al menos reducir uno o más de los problemas antes mencionados.

Esto se consigue mediante el método, el aparato, el programa informático y el producto de programa informático de las reivindicaciones adjuntas 1-13.

Gracias a ello, el signo de corrimiento y la magnitud de corrimiento se codifican por separado en los datos de vídeo, el dispositivo de control de filtrado es capaz de procesar al menos una porción de la primera imagen en el procedimiento de filtrado adaptativo de muestra mientras se aplica el corrimiento adaptativo de muestra, que ha sido determinado basándose en la magnitud de corrimiento y el signo de corrimiento. La codificación separada de la magnitud de corrimiento y del signo de corrimiento permite que la magnitud de corrimiento tanto para el corrimiento de borde como para el corrimiento de banda pueda codificarse y/o decodificarse mientras se utilizan las mismas capacidades de equipo físico/equipo lógico del dispositivo de control de filtrado. Por lo tanto, las descripciones, en términos de especificaciones estándar y los elementos de diseño de equipo físico/equipo lógico de la magnitud de corrimiento y del signo de corrimiento pueden ser menos complejas cuando se considera tanto el corrimiento de borde como el corrimiento de banda. Por lo tanto, se consigue un manejo más eficiente del corrimiento adaptativo de muestra. Como consecuencia, se obtiene una mejora del procedimiento de filtrado adaptativo de muestra en términos de eficiencia de codificación del corrimiento adaptativo de muestra.

## Breve descripción de los dibujos

Los diversos aspectos de las realizaciones divulgadas en el presente documento, incluyendo particularidades y ventajas de la misma, se entenderán fácilmente a partir de la siguiente descripción detallada y los dibujos adjuntos, en los que:

la figura 1 es un diagrama de bloques esquemático de un sistema de comunicación de radio de ejemplo, en el que se pueden implementar realizaciones de ejemplo,

la figura 2 es un diagrama de flujo esquemático que ilustra los métodos de ejemplo de acuerdo con las realizaciones del presente documento,

la figura 3 es otro diagrama de flujo esquemático que ilustra los métodos de ejemplo de acuerdo con las realizaciones del presente documento,

la figura 4 es un diagrama de bloques esquemático que ilustra un codificador de ejemplo que comprende un dispositivo de control de filtrado de acuerdo con las realizaciones del presente documento,

la figura 5 es un diagrama de bloques esquemático que ilustra un decodificador de ejemplo que comprende un dispositivo de control de filtrado de acuerdo con las realizaciones del presente documento,

la figura 6 es un diagrama de bloques esquemático que ilustra un equipo de usuario de ejemplo,

la figura 7 es otro diagrama de bloques esquemático que ilustra otro equipo de usuario de ejemplo,

la figura 8 es un diagrama de bloques esquemático adicional que ilustra un dispositivo de control de filtrado de ejemplo de acuerdo con las realizaciones del presente documento, y

la figura 9 es un diagrama de bloques esquemático adicional que ilustra un ordenador de ejemplo configurado de acuerdo con las realizaciones del presente documento.

## Descripción detallada

A lo largo de la siguiente descripción se han utilizado números de referencia similares para designar elementos, nodos de red, dispositivos, partes, elementos o características similares, cuando sea aplicable. En las figuras, las características que aparecen en algunas realizaciones se indican mediante líneas de trazos.

Con el fin de facilitar la comprensión de HEVC una descripción detallada de SAO en HVEC, de acuerdo con "WD6: Versión de trabajo 6 de codificación de vídeo de alta eficiencia ", JCTVC-H1003\_dk, pp 29-31, 34-35, 43-44, 70-72, 76, 83, 154-156, 170-171, 177, 186, Estados Unidos, febrero de 2012, se proporciona a continuación.

### Corrimientos adaptativos de muestra en versión HEVC

En HEVC SAO puede ser de dos tipos, ya sea corrimientos de borde o corrimientos de banda. Los corrimientos de borde pueden utilizarse en una de 4 direcciones, por ejemplo horizontal, vertical, diagonal desde la parte superior izquierda a la inferior derecha o diagonal desde la parte inferior izquierda hasta la parte superior derecha. Los cuatro tipos diferentes se muestran en la figura 1. La dirección específica está determinada por `sao_type_idx = 1..4`. `sao_type_idx = 5` se utiliza para corrimientos de banda.

Cuando se selecciona el corrimiento de banda (por ejemplo, `sao_type_idx` es 5) se utilizan cuatro corrimientos para cuatro bandas consecutivas. La posición de banda para la primera banda está indicada por `sao_band_position`. Se aplican los siguientes pasos ordenados:

Si `cldx` es igual a 0, la variable `bandShift` se establece igual a `BitDepthY - 5`

De lo contrario (`cldx` no es igual a 0), `bandShift` se establece igual a `BitDepthC - 5`. `cldx` corresponde al componente de color (YUV), por ejemplo `luma` o uno de los componentes de croma.

La variable `saoLeftClass` se establece igual a `sao_band_position[cldx][rx][ry]`. `rx` y `ry` identifican la posición actual de bloque de árbol de codificación.

El vector `bandTable` se define con 32 elementos y todos los elementos se establecen inicialmente en 0. A continuación, se modifican 4 de sus elementos (que indican la posición de las bandas para las que se transmite de forma explícita un corrimiento) de la siguiente manera:

```
for(i = 0; i < 4; i++)
{
    bandTable[(i + saoLeftClass) & 0x1F] = i + 1
}
```

El búfer de imagen reconstruido `recSaoPicture` se modifica de la siguiente manera.

```
bandIdx = ( recPicture[ xC + i, yC + j ] >> bandShift )
```

```
recSaoPicture[ xC + i, yC + j ] = recPicture[ xC + i, yC + j ] + saoValueArray[
bandTable[ bandIdx ] ]
```

con  $i = 0..nS-1$  y  $j = 0..nS-1$

Nota: para hacer que el valor filtrado de la banda SAO permanezca dentro del intervalo permitido de acuerdo con la profundidad de bit se necesita un accionamiento de recorte. Esto no se especifica en el borrador de trabajo HEVC, pero existe un recorte basado en la búsqueda de tablas en el equipo lógico de referencia HEVC.

Cuando se seleccionan corrimientos de borde (por ejemplo, `sao_type_idx` es 1 ó 2 ó 3 ó 4) se utilizan cuatro corrimientos para tipos de borde específicos. Estos cuatro tipos se ilustran en la figura siguiente. Los tipos de borde se derivan para cada píxel por:

$$\text{edgeIdx} = 2 + \sum_k \text{Sign}(\text{recPicture}[x, y] - \text{recPicture}[x + \text{hPos}[k], y + \text{vPos}[k]]) \quad \text{con } K = 0 \dots 1$$

donde `recPicture` es la imagen después de desbloquear el proceso de filtro, `hPos` y `vPos` se definen a continuación:

Tabla I: Especificación de `hPos` y `vPos` de acuerdo con el tipo de proceso de corrimiento adaptativo de muestra

<code>sao_type_idx [cldx] [rx] [ry]</code>	11	22	33	44
<code>hPos[0]</code>	-- 1	00	-- 1	11
<code>hPos[1]</code>	11	10	11	-- 1
<code>vPos[0]</code>	00	-- 1	-- 1	-- 1
<code>vPos[1]</code>	00	11	11	11

El búfer de imagen reconstruido se modifica como:

$$\text{recSaoPicture}[x_C + i, y_C + j] = \text{recPicture}[x_C + i, y_C + j] + \text{saoValueArray}[\text{edgeTable}[\text{edgeIdx}]]$$

con  $i = 0..n_S - 1$  y  $j = 0..n_S - 1$  donde `edgeTable[5] = {1, 2, 0, 3, 4}`.

Nota: para hacer que el valor filtrado de borde SAO se mantenga dentro del intervalo permitido de acuerdo con la profundidad de bit, se necesita un accionamiento de recorte. Esto no se especifica en el borrador de trabajo HEVC pero existe recorte en equipo lógico.

Tabla II: Cuatro tipos de borde para corrimiento de borde

	<code>edgeIdx = 0</code>
	<code>edgeIdx = 1</code>
	<code>edgeIdx = 3</code>
	<code>edgeIdx = 4</code>

`saoValueArray` se establece igual a `SaoOffsetVal [cldx] [rx] [ry]` que se define a continuación.

`sao_type_idx [cldx] [rx] [ry]` indica el tipo de corrimiento como se especifica en la Tabla 7-6 del bloque de árbol actual de codificación en la posición `rx` y `ry` para el componente de color `cldx`.

Cuando `sao_type_idx [cldx] [rx] [ry]` no está presente, se deduce de la siguiente manera.

Si `sao_merge_up_flag` es igual a 1, `sao_type_idx [cldx] [rx] [ry]` se establece igual a `sao_type_idx [cldx] [rx] [ry - 1]`.

De lo contrario, `sao_type_idx [cldx] [rx] [ry]` se establece igual a `sao_type_idx [cldx] [rx - 1] [ry]`.

Tabla III: Especificación del tipo de borde para SAO

sao_type_idx [cldx] [rx] [ry]	Tipo de borde (informativo)
0	No aplicado
1	Borde de 0 grados 1D
2	Borde de 90 grados 1D
3	Borde de 135 grados 1D
4	Borde de 45 grados 1D
5	Banda

5 sao\_band\_position [cldx] [rx] [ry] indica el desplazamiento del corrimiento de banda del intervalo de píxeles cuando sao\_type\_idx [cldx] [rx] [ry] es igual a 5.

Cuando sao\_band\_position [cldx] [rx] [ry] no está presente se deduce lo siguiente.

10 Si sao\_merge\_up\_flag es igual a 1, sao\_band\_position [cldx] [rx] [ry] se establece igual a sao\_band\_position [cldx] [rx] [ry - 1].

De lo contrario, sao\_band\_position [cldx] [rx] [ry] se establece igual a sao\_band\_position [cldx] [rx - 1] [ry].

15 sao\_offset [cldx] [rx] [ry] [i] indica el valor de corrimiento de la i-ésima categoría de bloque de árbol de codificación actual en la posición rx y ry para el componente de color cldx.

La variable bitDepth se deriva de la siguiente manera.

20 Si cldx es igual a 0, bitDepth se establece igual a BitDepthY..

De lo contrario (cldx es igual a 1 ó 2), bitDepth se establece igual a BitDepthC.

25 Es un requisito de la conformidad de flujo de bits que cuando sao\_type\_idx [cldx] [rx] [ry] no es igual a 5, los valores de sao\_offset [cldx] [rx] [ry] [i] estarán en el intervalo de 0 a  $(1 \ll (\text{Min}(\text{bitDepth}, 10) - 5)) - 1$ , inclusive y de lo contrario, los valores de sao\_offset [cldx] [rx] [ry] [i] estarán en el intervalo de  $-(1 \ll (\text{Min}(\text{bitDepth}, 10) - 5))$  a  $(1 \ll (\text{Min}(\text{bitDepth}, 10) - 5)) - 1$ , inclusive.

Cuando sao\_offset [cldx] [rx] [ry] [i] no está presente, se deduce lo siguiente.

30 Si sao\_merge\_up\_flag es igual a 1, sao\_offset [cldx] [rx] [ry] [i] se establece igual a sao\_offset [cldx] [rx] [ry - 1] [i].

De lo contrario, sao\_offset [cldx] [rx] [ry] [i] se establece igual a sao\_offset [cldx] [rx - 1] [ry] [i].

35 La variable offsetSign se deriva de la siguiente manera.

Si sao\_type\_idx [cldx] [rx] [ry] es menor que 5 e i es mayor que 1, offsetSign se establece en -1.

De lo contrario (sao\_type\_idx [cldx] [rx] [ry] es igual a 5 o i es menor que 2), offsetSign se establece en 1.

40 La matriz SaoOffsetVal se deriva de la siguiente manera.

$$\text{SaoOffsetVal}[\text{cldx}][\text{rx}][\text{ry}][0] = 0$$

$$\text{SaoOffsetVal}[\text{cldx}][\text{rx}][\text{ry}][i + 1] =$$

$$\text{offsetSign} * \text{sao\_offset}[\text{cldx}][\text{rx}][\text{ry}][i] \ll (\text{bitDepth} - \text{Min}(\text{bitDepth}, 10))$$

Donde rx y ry declaran una posición dentro del corte/imagen y cldx es el componente de color respectivo.

La figura 1 representa un sistema 100 de comunicaciones por radio de ejemplo en el que se pueden implementar realizaciones del presente documento. En este ejemplo, el sistema 100 de comunicaciones por radio es un sistema LTE (evolución a largo plazo). En otros ejemplos, el sistema de comunicación por radio puede ser cualquier red basada en radio o cualquier sistema de comunicación celular 3GPP, tal como una red de acceso múltiple por división de código de banda ancha (WCDMA), un sistema global para comunicaciones móviles (red GSM) o similares.

El sistema 100 de comunicación por radio comprende un dispositivo 110 de red. El dispositivo 110 de red puede tener forma o estar compuesto de una estación base de radio, un Nodo-B, un nodo de red de radio, un controlador de red de radio, un controlador de estación base o cualquier otro nodo de red en el sistema 100 de comunicación por radio.

El dispositivo 110 de red puede convertir vídeo de acuerdo con un estándar de codificación de vídeo a otro estándar de codificación de vídeo, por ejemplo, si se ha establecido que un equipo 120 de usuario receptor sólo es capaz de o prefiere otro estándar de codificación de vídeo que el enviado desde una unidad 130 de envío. El equipo 120 de usuario receptor y la unidad 130 de envío están comprendidos en el sistema 100 de comunicación por radio. Tal como se utiliza en el presente documento, el término "equipo de usuario" y "unidad de envío" pueden referirse a un teléfono móvil, un teléfono celular, un asistente personal digital (PDA) equipado con capacidades de comunicación por radio, un teléfono inteligente, un portátil u ordenador personal (PC) equipado con un módem de banda ancha móvil interno o externo, una tableta PC con capacidades de comunicación por radio, un dispositivo electrónico portátil de comunicación por radio, un dispositivo de sensor equipado con capacidades de comunicación por radio o similar. El sensor puede ser cualquier tipo de sensor meteorológico, como viento, temperatura, presión de aire, humedad, etc. Como ejemplos adicionales, el sensor puede ser un sensor de luz, un interruptor electrónico, un micrófono, un altavoz, un sensor de cámara, etc.

Además, el dispositivo 110 de red puede comprender un codificador y/o decodificador 140. Con más detalle, el codificador y/o decodificador 140 pueden comprender un dispositivo 150 de control de filtrado, en el que pueden realizarse realizaciones del método ilustrado en la figura 2, como se describe a continuación.

La figura 2 ilustra un método de ejemplo, realizado por el dispositivo 150 de control de filtrado, para procesar una primera imagen en un procedimiento de filtrado adaptativo de muestra. Esto significa que la primera imagen es procesada por el dispositivo 150 de control de filtrado de acuerdo con el procedimiento de filtrado adaptativo de muestra. La primera imagen se reconstruye a partir de datos de vídeo, por ejemplo en una acción previa de acuerdo con métodos de reconstrucción conocidos.

Como se utiliza en el presente documento, el término "una segunda imagen" se puede utilizar para expresar que la primera imagen, o al menos una porción de la misma, se ha procesado en el procedimiento de filtrado adaptativo de muestra. La segunda imagen se puede obtener como una imagen de tamaño completo que es una versión modificada de la primera imagen. La segunda imagen se puede obtener como una versión modificada de la primera imagen reemplazando valores de imagen en la primera imagen directamente si se mantiene una copia de la porción de la primera imagen. La copia de la porción de la primera imagen no modificada se puede utilizar para determinar la categoría de borde en caso de corrimientos de borde. Alternativamente o adicionalmente, la segunda imagen se puede obtener almacenando los valores de píxel modificados en una matriz adicional y, a continuación, cuando se completa el procesamiento de la primera imagen, los valores de píxel modificados de la matriz adicional pueden ser copiados en un búfer de la primera imagen. Por lo tanto, convirtiendo la primera imagen en la segunda imagen.

Las siguientes acciones se pueden realizar en cualquier orden adecuado.

#### Acción 201

Con el fin de poder procesar la primera imagen en la acción 207 de acuerdo con el procedimiento de filtrado adaptativo de muestra mientras se aplica un corrimiento adaptativo de muestra, el dispositivo 150 de control de filtrado obtiene una magnitud de corrimiento de los datos de vídeo. La magnitud de corrimiento indica una magnitud del corrimiento adaptativo de muestra a aplicar en el procedimiento de filtrado adaptativo de muestra, por ejemplo durante el procesamiento de la primera imagen en la acción 207.

De acuerdo con algunas primeras realizaciones, el dispositivo 150 de control de filtrado puede obtener la magnitud de corrimiento estableciendo, por ejemplo comprendiendo la acción de ajuste, la magnitud de corrimiento basada en la profundidad de bits obtenida a partir de los datos de vídeo. Otras acciones con referencia a las primeras realizaciones se describen a continuación.

De acuerdo con algunas segundas realizaciones, el dispositivo 150 de control de filtrado puede obtener la magnitud de corrimiento obteniendo, por ejemplo comprendiendo la acción de obtener, un valor booleano a partir de los datos de vídeo. A continuación, el dispositivo 150 de control de filtrado puede derivar la magnitud de corrimiento basada en la profundidad de bits obtenida a partir de los datos de vídeo cuando el valor booleano es verdadero.

#### Acción 202

El dispositivo 150 de control de filtrado puede obtener un valor de los datos de vídeo. El valor se puede utilizar en la acción 203 directamente debajo.

#### Acción 203

De acuerdo con algunas terceras realizaciones, el dispositivo 150 de control de filtrado puede mapear el valor, obtenido en la acción 202, con la magnitud de corrimiento de una manera no lineal. El mapeo puede ser cada vez más escaso, en términos de resolución para la magnitud de corrimiento obtenida, cuanto mayor sea la magnitud de corrimiento.

#### Acción 204

El dispositivo de control de filtrado 150 obtiene un signo de corrimiento de los datos de vídeo. El signo de corrimiento indica un signo del corrimiento adaptativo de muestra.

De acuerdo con las primeras realizaciones, el dispositivo 150 de control de filtrado puede obtener el signo de corrimiento estableciendo el signo de corrimiento basándose en la profundidad de bits. De esta manera, un valor de píxel de la segunda imagen, por ejemplo después del procesamiento en el procedimiento de filtrado adaptativo de muestra, está entre 0 y un valor máximo de acuerdo con la profundidad de bits obtenida después del procesamiento 207 de al menos dicha porción de la primera imagen.

#### Acción 205

El dispositivo 150 de control de filtrado determina el corrimiento adaptativo de muestra basado en la magnitud de corrimiento y el signo de corrimiento.

De acuerdo con algunas cuartas realizaciones, el dispositivo 150 de control de filtrado determina el corrimiento adaptativo de muestra calculando directamente, por ejemplo comprendiendo la acción de calcular directamente el corrimiento adaptativo de muestra por medio de accionamientos de signo y accionamientos de cambio en la magnitud de corrimiento y el signo de corrimiento.

Se puede obtener una magnitud de cambio del accionamiento de cambio a partir de los datos de vídeo. Alternativamente o adicionalmente, la magnitud de cambio del accionamiento de cambio puede estar predefinida. Por ejemplo, una magnitud de cambio predefinida puede ser anulada por una magnitud de cambio obtenida a partir de los datos de vídeo. La magnitud de cambio indica un número de bits a desplazar en el accionamiento de cambio. Como ejemplo,  $1 < x$ , donde  $x$  es la magnitud de cambio, significa que los bits se desplazan  $x$  pasos hacia la izquierda. Esto corresponde a una multiplicación por  $2^x$ . Por lo tanto, si la magnitud de cambio es 1, el accionamiento de cambio corresponde a una multiplicación por 2. De manera similar, si la magnitud de cambio es 2, el accionamiento de cambio corresponde a una multiplicación por 4 y así sucesivamente.

#### Acción 206

El dispositivo 150 de control de filtrado puede obtener, a partir de los datos de vídeo, una posición de banda que indica una primera banda de un conjunto de bandas consecutivas. La magnitud de corrimiento y el signo de corrimiento pueden aplicarse a una o más del conjunto de bandas consecutivas.

#### Acción 207

El dispositivo 150 de control de filtrado procesa, en el procedimiento de filtrado adaptativo de muestra, al menos una porción de la primera imagen mientras se aplica el corrimiento adaptativo de muestra. De esta manera, tal como se ha explicado anteriormente, se puede obtener la segunda imagen. La segunda imagen puede mejorarse gracias al procesamiento de acuerdo con el procedimiento de filtrado adaptativo de muestra.

De acuerdo con las primeras realizaciones, el dispositivo 150 de control de filtrado procesa al menos una porción de la primera imagen aplicando, por ejemplo comprendiendo la acción de aplicar, el corrimiento adaptativo de muestra a una de las bandas consecutivas. El signo de corrimiento puede ser positivo, por ejemplo restringido a ser positivo: cuando la posición de banda indica una banda inicial entre bandas que cubren valores de píxeles posibles de acuerdo con la profundidad de bits obtenida a partir de los datos de vídeo para la segunda imagen. Además, el signo de corrimiento puede ser negativo, por ejemplo restringido a ser negativo: cuando la posición de banda indica una banda para la cual el corrimiento adaptativo de muestra cubre una última banda entre bandas que cubren valores de píxeles posibles de acuerdo con la profundidad de bits obtenida a partir de los datos de vídeo para la segunda imagen.

De acuerdo con algunas realizaciones, el número de corrimientos de banda para SAO, cuando se aplica en la



codificación de imagen o video, se reduce. Por lo tanto, la memoria consumida para almacenar el corrimiento de banda puede reducirse también y también mejorar la eficiencia de codificación.

Gracias al uso de un ancho diferente de la banda, el corrimiento puede aplicarse a más de la precisión de la posición de la banda. En otras palabras, cuando un ancho de la banda es diferente, es decir, más ancha, que la resolución, o exactitud, de la posición de banda, el corrimiento adaptativo de muestra puede aplicarse a muchos valores de píxeles mientras que todavía tiene una alta precisión con respecto a una posición de arranque, por ejemplo la posición de banda, donde puede aplicarse el corrimiento adaptativo de muestra.

El método puede realizarse en el dispositivo 150 de filtrado que puede estar localizado en un codificador o un decodificador como se ilustra esquemáticamente en las figuras 4 y 5. Las figuras 4 y 5 ilustran el ejemplo cuando el método se realiza dentro del bucle de codificación.

Una ventaja con algunas realizaciones es que se puede utilizar un corrimiento de banda al mismo tiempo que se obtiene una eficiencia de codificación similar a la de 4 corrimientos de banda.

En esta realización se supone que cada banda tiene 8 valores de píxeles de ancho (0..7, 8..15, ...) igual que HEVC y para una profundidad de 8 bits esto corresponde a 32 bandas. Otros anchos de bandas son posibles, por ejemplo para una profundidad de 10 bits cada banda tiene normalmente 32 valores de píxeles de ancho.

La posición de banda `sao_band_position` determina la posición de la primera banda a la que se debe aplicar un corrimiento.

Se aplica el mismo corrimiento tanto para la banda señalada por la posición de la primera banda como también para una o más bandas vecinas.

Se decodifica una posición de banda `sao_band_position`, `saoLeftClass`. Se decodifica al menos un corrimiento. El corrimiento se establece para el índice correspondiente de una tabla, por ejemplo `bandTable` abajo, incluyendo todas las bandas. Otro índice de la tabla tiene un corrimiento cero.

En el ejemplo que se muestra a continuación sólo se utiliza un corrimiento ( $n = 1$ ) y dos bandas tienen el mismo corrimiento, la banda señalada por la posición de banda (`saoLeftClass`) y la banda señalada por `saoLeftClass-1`. Sin embargo, podría especificarse como la banda señalada por la posición de banda `saoLeftClass` y la banda señalada por `saoLeftClass + 1`.

```
for(i = 0; i < n; i++)
{
    bandTable[(i + saoLeftClass) & 0x1F] = i + 1
    if(saoLeftClass > 0)
    {
        bandTable[(i + saoLeftClass - 1) & 0x1F] = i + 1
    }
}
```

El búfer de imagen reconstruido `recSaoPicture` se modifica de la siguiente manera.

$$\text{bandIdx} = (\text{recPicture}[\text{xC} + \text{i}, \text{yC} + \text{j}] \gg \text{bandShift})$$

```
recSaoPicture[xC + i, yC + j] = recPicture[xC + i, yC + j] + saoValueArray[
bandTable[bandIdx]]
```

con  $i = 0..nS-1$  y  $j = 0..nS-1$

La variable `offsetSign` se deriva de la siguiente manera.

Si `sao_type_idx[cldx][rx][ry]` es menor que 5 e  $i$  es mayor que 1, `offsetSign` se establece en -1.

De lo contrario (`sao_type_idx[cldx][rx][ry]` es igual a 5 o  $i$  es menor que 2), `offsetSign` se establece en 1.

La matriz `SaoOffsetVal` se deriva de la siguiente manera cuando `sao_type_idx[cldx][rx][ry]` es igual a 5.

$$\text{SaoOffsetVal}[\text{cldx}][\text{rx}][\text{ry}][0] = 0$$

$$\text{SaoOffsetVal}[\text{cldx}][\text{rx}][\text{ry}][1] = \text{sao\_offset\_sign}[\text{cldx}][\text{rx}][\text{ry}](1 \ll \text{sao\_offset}[\text{cldx}][\text{rx}][\text{ry}])$$

Tabla V: Sintaxis VLC de corrimiento adaptativo de muestra

<code>sao_offset_vlc( rx, ry, cldx ) {</code>	<b>Descriptor</b>
<code>sao_type_idx[ cldx ][ rx ][ ry ]</code>	<code>ue(v)</code>
<code>if( sao_type_idx[ cldx ][ rx ][ ry ] == 5 ) {</code>	
<code>sao_band_position[ cldx ][ rx ][ ry ]</code>	<code>u(5)</code>
<code>sao_offset_sign[ cldx ][ rx ][ ry ]</code>	<code>u(1)</code>
<code>sao_offset[ cldx ][ rx ][ ry ]</code>	<code>ue(v)</code>
<code>} else if( sao_type_idx[ cldx ][ rx ][ ry ] != 0 )</code>	
<code>for( i = 0; i &lt; 4; i++ )</code>	
<code>sao_offset[ cldx ][ rx ][ ry ][ i ]</code>	<code>ue(v)</code>
<code>}</code>	

5

Tabla VI: Sintaxis CABAC de corrimiento adaptativo de muestra

<code>sao_offset_cabac( rx, ry, cldx ) {</code>	<b>Descriptor</b>
<code>sao_type_idx[ cldx ][ rx ][ ry ]</code>	<code>ae(v)</code>
<code>if( sao_type_idx[ cldx ][ rx ][ ry ] == 5 ){</code>	
<code>sao_band_position[ cldx ][ rx ][ ry ]</code>	<code>ae(v)</code>
<code>sao_offset_sign[ cldx ][ rx ][ ry ]</code>	<code>ae(v)</code>
<code>sao_offset[ cldx ][ rx ][ ry ]</code>	<code>ae(v)</code>
<code>} else if( sao_type_idx[ cldx ][ rx ][ ry ] != 0 )</code>	
<code>for( i = 0; i &lt; 4; i++ )</code>	
<code>sao_offset[ cldx ][ rx ][ ry ][ i ]</code>	<code>ae(v)</code>
<code>}</code>	

El subrayado indica cambios en comparación con versiones anteriores de la presente especificación estándar.

10

`sao_offset_sign[ cldx ][ rx ][ ry ]` indica el signo del valor de corrimiento de banda de la *i*-ésima categoría de bloque de árbol de codificación actual en la posición *rx* y *ry* para el componente de color *cldx*

Una extensión de este algoritmo es variar el número de bandas al que se aplica un corrimiento. El número de

bandas puede ser 1, 2, 3 ó 4. El número de bandas puede ser señalado con código de longitud variable o código de longitud fija o codificado por CABAC.

Con referencia a la figura 2, se ha descrito un conjunto de realizaciones. El conjunto de realizaciones incluye las primeras realizaciones, las segundas realizaciones, las terceras realizaciones y las cuartas realizaciones. Las realizaciones primera a cuarta se pueden utilizar por separado o pueden combinarse. Se observará, de hecho, que estas realizaciones también pueden realizarse sin las acciones 201, 204 y 205 en las que se utiliza el signo de corrimiento y la magnitud de corrimiento separadas para la determinación del corrimiento adaptativo de muestra.

Las realizaciones pueden utilizarse en un codificador y/o un decodificador o completamente fuera del bucle de codificación como un filtro de poste. En las aplicaciones típicas, las realizaciones se aplican después del desbloqueo de filtrado si se utiliza dicho filtrado. Sin embargo, en alguna aplicación puede ser de interés aplicarlo antes del desbloqueo.

Con referencia a la figura 3, se describen las realizaciones primera a cuarta.

Empezando por las primeras realizaciones, se evitan los accionamientos de recorte en los corrimientos de borde o banda gracias a una limitación del corrimiento adaptativo de muestra, de manera que no es necesario el recorte. Ventajosamente, se puede reducir la complejidad del codificador y del decodificador.

Los corrimientos de banda, por ejemplo un corrimiento adaptativo de muestra para una banda en particular, se pueden restringir sólo permitiendo corrimientos positivos para la primera banda y evitar tener mayores corrimientos negativos para otras bandas que pueden causar la necesidad de accionamiento de recorte. De forma similar para la banda más alta sólo permiten los corrimientos negativos y para las otras bandas se requiere que los corrimientos positivos no provoquen que el valor de píxel filtrado por SAO exceda el valor máximo de acuerdo con la profundidad de bits.

Tanto los corrimientos de banda como de borde pueden estar restringidos para la LCU actual, de modo que un corrimiento seleccionado no puede causar que el valor de píxel filtrado por SAO descienda por debajo de 0 o exceda el valor máximo de acuerdo con la profundidad de bits.

El método se realiza en un dispositivo de control de filtrado que puede estar situado en un codificador o un decodificador como se ilustra esquemáticamente en las figuras 4 y 5. Las figuras 4 y 5 ilustran el ejemplo cuando el método se realiza dentro del bucle de codificación.

Una ventaja adicional con las primeras realizaciones es que un decodificador no necesita hacer accionamiento de recorte basado en píxeles. Como resultado, como se mencionó anteriormente, la complejidad del decodificador se reduce.

Ahora, se presentan algunos detalles adicionales con respecto a las primeras realizaciones. Las primeras realizaciones consiguen una limitación de corrimientos de borde o banda para evitar los accionamientos de recorte.

Una restricción que puede aplicarse tanto al corrimiento del borde como al corrimiento de banda es especificar que un flujo de bits sea conforme al estándar que ningún corrimiento transmitido puede hacer que el píxel filtrado por SAO exceda el valor máximo o el valor mínimo de acuerdo con la profundidad de bits. Por lo tanto, no se debe especificar ningún accionamiento de recorte para los píxeles filtrados por SAO en el estándar y el decodificador se ahorrará de tales accionamientos para el filtrado SAO.

Ejemplo: supongamos que se utiliza vídeo de 8 bits. En ese caso, el valor mínimo es 0 y el valor máximo es 255. Siguiendo esta restricción, el valor de píxel filtrado por SAO estará en el intervalo de 0 a 255.

Una restricción para el corrimiento de banda es sólo permitir el corrimiento positivo para la primera banda y el corrimiento negativo para la última banda. Otras bandas pueden estar restringidas para tener un corrimiento máximo y mínimo para que el filtrado SAO con corrimiento de banda permanezca dentro del intervalo de profundidad de bits.

Ejemplo: supongamos que el bandIdx es 0 para que el píxel actual sea filtrado por corrimiento de banda. En ese caso, el corrimiento será siempre positivo para que el valor de píxel filtrado por SAO esté dentro del intervalo de 0 hasta el valor máximo de acuerdo con la profundidad de bit.

Una extensión adicional de esta restricción de sólo corrimiento de banda es omitir la señalización de cualquier signo para la primera y última banda y, por tanto, eliminar algunos gastos generales al mismo tiempo que evita el accionamiento de recorte después del filtrado SAO para asegurar que está dentro del intervalo de profundidad de bits.

TABLA VII: Ejemplo:

sao_offset_vlc( rx, ry, cIdx ) {	<b>Descriptor</b>
sao_type_idx[ cIdx ][ rx ][ ry ]	ue(v)
if( sao_type_idx[ cIdx ][ rx ][ ry ] == 5 ) {	
sao_band_position[ cIdx ][ rx ][ ry ]	u(5)
for( i = 0; i < 4; i++ ){	
If((sao_band_position==0 && i==0)    Sao_band_position+i==31))	
sao_offset[ cIdx ][ rx ][ ry ][ i ]	ue(v)
Else	
sao_offset[ cIdx ][ rx ][ ry ][ i ]	se(v)
}	
} else if( sao_type_idx[ cIdx ][ rx ][ ry ] != 0 )	
for( i = 0; i < 4; i++ )	
sao_offset[ cIdx ][ rx ][ ry ][ i ]	ue(v)
}	

An array SaoOffsetVal is specified as

```

SaoOffsetVal[cIdx][rx][ry][0] = 0
if ((sao_type_idx[cIdx][rx][ry]==5)&&(i==3)&&(
sao_band_position[cIdx][rx][ry]+i==31)
SaoOffsetVal[cIdx][rx][ry][i+1] =
-(sao_offset[cIdx][rx][ry][i]<<(bitDepth
- Min(bitDepth, 10)))
else
SaoOffsetVal[cIdx][rx][ry][i + 1] =
sao_offset[cIdx][rx][ry][i]<<(bitDepth-
Min(bitDepth, 10))
with i = 0..3

```

- 5 Con referencia de nuevo a las segundas realizaciones, de acuerdo con las cuales se activa/desactiva un corrimiento fijo para un corrimiento de borde respectivo. El corrimiento fijo se ajusta de acuerdo con la profundidad de bits. Por lo tanto, esta realización se refiere a la restricción del corrimiento del borde a ser 0 o tener un valor predeterminado de acuerdo con la profundidad de bit. Ventajosamente, se reduce una sobrecarga para codificar corrimientos de borde y consumo de memoria para almacenarlos. Otra ventaja es que el decodificador no necesita realizar ningún
- 10 accionamiento de recorte basado en píxeles después de filtrar con corrimientos de borde.

Estas realizaciones se refieren a la restricción del corrimiento del borde a ser 0 o tener un valor predeterminado de acuerdo con la profundidad de bit.

- 15 En este caso sólo se señala si se utiliza un corrimiento de borde o no se utiliza como se indica por un indicador sao\_offset [cIdx] [rx] [ry] [i] para el corrimiento de borde respectivo. La bandera es 1 ó 0.

edgIdx igual a 0 tienen SaoOffsetVal igual a sao\_offset [cIdx] [rx] [ry] [i]<<(bitDepth-8)

- 20 edgIdx igual a 1 tienen SaoOffsetVal igual a sao\_offset [cIdx] [rx] [ry] [i]<<(bitDepth-8)

edgIdx igual a 2 tienen SaoOffsetVal igual a 0

edgIdx igual a 3 tienen SaoOffsetVal igual a -(

- 25 Sao\_offset [cIdx] [rx] [ry] [i]<<(bitDepth-8))

edgIdx igual a 4 tienen SaoOffsetVal igual a -(

Sao\_offset [cldx] [rx] [ry] [i] << (bitDepth-8))

Un beneficio adicional de estas realizaciones es que no se necesitará recorte en el proceso de filtrado SAO para vídeo de 8 bits (bitDepth igual a 8) puesto que el valor de píxel filtrado por SAO no excederá el valor de píxel vecino.

La matriz SaoOffsetVal se deriva de la siguiente manera cuando

sao\_type\_idx [cldx] [rx] [ry] está en el intervalo de 1 a 4:

$$\text{SaoOffsetVal}[\text{cldx}][\text{rx}][\text{ry}][0] = 0$$

$$\text{SaoOffsetVal}[\text{cldx}][\text{rx}][\text{ry}][i+1] =$$

$$\text{offsetSign} * (\text{sao\_offset}[\text{cldx}][\text{rx}][\text{ry}][i] << (\text{bitDepth}-8)) \quad (7-37)$$

Donde offsetSign es definido anteriormente.

Para permitir el filtrado SAO con corrimientos de borde para una profundidad de bits mayor que 8 sin recorte, la clasificación del tipo de borde se puede modificar de la siguiente manera:

Los tipos de borde se derivan para cada píxel mediante:

$$\text{edgeIdx} = 2 + \sum_k \text{Sign}(\text{recPicture}[x, y] >> \text{bitDepthIncr} - \text{recPicture}[x + \text{hPos}[k], y + \text{vPos}[k]] >> \text{bitDepthIncr})) \quad \text{con } k = 0 \dots 1$$

donde recPicture es la imagen después del proceso de filtro de desbloqueo, hPos y vPos se definen anteriormente. BitDepthIncr es igual a bitDepth-8.

Para permitir el filtrado SAO con corrimientos de borde con corrimientos de borde mayores y el trabajo sin recorte de un corrimiento (rightShift) que se determina por el corrimiento de borde se puede utilizar de la siguiente manera:

Los tipos de borde se derivan para cada píxel por:

$$\text{edgeIdx} = 2 + \sum_k \text{Sign}(\text{recPicture}[x, y] >> \text{rightShift} - \text{recPicture}[x + \text{hPos}[k], y + \text{vPos}[k]] >> \text{rightShift})) \quad \text{con } k = 0 \dots 1$$

Si el corrimiento de borde es 3 rightShift es igual a 2. Normalmente, rightShift puede ajustarse a la potencia de 2 que esté más cerca de la magnitud de corrimiento máxima.

Con referencia de nuevo a las terceras realizaciones, de acuerdo con las cuales una sobrecarga para codificar corrimientos con grandes magnitudes para SAO, cuando se aplica en codificación de imagen o vídeo, puede reducirse. Por lo tanto, se pueden conseguir mejoras con respecto a la eficiencia de codificación.

Estas realizaciones pueden aplicarse tanto a los corrimientos de borde como a los corrimientos de banda

Después de que los valores de píxeles reconstruidos hayan sido filtrados por desbloqueo. Esto se hace tanto en el lado del codificador como en el lado del decodificador. También es posible aplicarlo solamente para el corrimiento de banda o el corrimiento de borde.

De acuerdo con los corrimientos decodificados de terceras realizaciones, que normalmente son distribuidos linealmente con respecto a la magnitud de corrimiento, son mapeados con un orden no lineal de magnitudes de corrimiento. De este modo, los corrimientos de gran magnitud son más escasamente representados que los corrimientos con magnitudes menores, por ejemplo con base 2, por ejemplo, ..., -8, -4, -2, -1, 0, 1, 2, 4, 8, ..., etc. después del mapeado. De manera similar en el codificador, los corrimientos están restringidos a magnitudes de corrimiento no uniformemente distribuidas y son mapeados a una distribución lineal de magnitudes de corrimiento, por ejemplo, ..., -3, -2, -1, 0, 1, 2, 3, ..., etc., antes de la codificación. La codificación/decodificación de corrimiento

puede codificarse por codificación de longitud variable VLC, codificación de longitud fija FLC o CABAC. El método se realiza en un dispositivo de control de filtrado que puede estar localizado en un codificador o un decodificador como se ilustra esquemáticamente en las figuras 4 y 5. Las figuras 4 y 5 ilustran el ejemplo cuando el método se realiza dentro del bucle de codificación.

Una ventaja con las terceras realizaciones es que se pueden utilizar grandes corrimientos utilizando menos bits. Por lo tanto, la eficiencia de codificación se mejora.

Ahora, se presentan algunos detalles adicionales con respecto a las terceras realizaciones. Las terceras realizaciones consiguen una codificación y/o decodificación más eficiente de los corrimientos de banda y/o corrimientos de borde.

Estas realizaciones se pueden aplicar para codificar/decodificar corrimientos de banda y/o borde. Los corrimientos son mapeados entre magnitudes lineales y no lineales.

En esta descripción se supone que los corrimientos son codificados con una base dos pero pueden utilizarse otras bases o la búsqueda de tabla para encontrar la correspondencia entre el corrimiento codificado/decodificado y el corrimiento real que se va a utilizar en el proceso de filtrado SAO.

En el codificador un corrimiento SaoOffsetVal está restringido a... - 8, -4, -2, -1, 0, 1, 2, 4, 8...

El corrimiento a codificar se determina como:

$$\text{sao\_offset} = \text{table}(\text{abs}(\text{SaoOffsetVal})) * \text{sign}(\text{SaoOffsetVal})$$

donde la tabla corresponde a  $\log_2(\text{abs}(\text{SaoOffsetVal}))$  o la función aritmética real  $\log_2$  en este ejemplo y donde

$\text{Sign}(x)$  - cuando  $x > 0$  es igual a 1, cuando  $x = 0$  es igual a 0, de lo contrario -1

Un ejemplo de computación de signo rápido es desde HEVC donde

$$\text{Sign}(x) = ((x \gg 31) \mid ((\text{int})(((\text{unsigned int}) -$$

$x \gg n$  - corresponde a  $x/2^n$  por ejemplo si  $n = 1$  es igual a  $x/2$

$x \ll n$  - corresponde a  $x * 2^n$  por ejemplo, si  $n = 1$  es igual a  $x * 2$

$x \mid y$  - OR lógico, cuando al menos uno de  $x$  o  $y$  es 1 es igual a 1

$\log_2(x)$  - es el logaritmo con base 2 de  $x$

bit depth - bit depth define el intervalo de valores de píxeles permitidos. El contenido de 8 bits solo tiene valores entre 0 y 255 ( $2^8 - 1$ ).

El corrimiento se codifica a continuación mediante código de longitud variable o código de longitud fija o CABAC. La magnitud y el signo pueden ser codificados por separado o juntos.

El decodificador decodifica la palabra de código para obtener el corrimiento y luego determina SaoOffsetVal:

$$\text{SaoOffsetVal} = (1 \ll \text{abs}(\text{sao\_offset})) * \text{sign}(\text{sao\_offset})$$

En el caso de que sólo se codifique un corrimiento, se puede prohibir que SaoOffsetVal sea igual a 0 y así reducir el número de bits para codificar los corrimientos no nulos.

SaoOffsetVal se utiliza entonces para actualizar el valor de píxel actual cuando se ha seleccionado SAO para ser utilizado para ese píxel:

$$\text{recSaoPicture}[x\text{C} + i, y\text{C} + j] = \text{recPicture}[x\text{C} + i, y\text{C} + j] + \text{SaoOffsetVal}$$

Con referencia de nuevo a las cuartas realizaciones, en las que el corrimiento de banda se deriva mediante cálculo directo utilizando operaciones de signo. De esta manera, se omite una necesidad de búsqueda de tabla basada en píxeles, cuando se aplican corrimientos de banda. Por lo tanto, estas realizaciones pueden reducir ventajosamente

la memoria consumida para almacenar corrimientos de banda.

El método se realiza en un dispositivo de control de filtrado que puede estar localizado en un codificador o un decodificador como se ilustra esquemáticamente en las figuras 4 y 5. Las figuras 4 y 5 ilustran el ejemplo cuando el método se realiza dentro del bucle de codificación.

Una ventaja con las cuartas realizaciones es que los corrimientos de banda pueden ser implementados sin mirar la tabla.

Ahora, se presentan algunos detalles adicionales con respecto a las cuartas realizaciones.

En este ejemplo, se supone que sólo se transmite un corrimiento de banda.

Determine bandIdx para el píxel respectivo como:

**bandIdx = ( recPicture[ xC + i, yC + j ] >> bandShift )-saoLeftClass**

Modifique el píxel como:

**recSaoPicture[ xC + i, yC + j ] = recPicture[ xC + i, yC + j ] + (sign(bandIdx  
+R1)+sign(-bandIdx+R2)) << C**

donde C puede tener un valor fijo determinado por el estándar o una longitud variable o longitud fija codificada y proporcionada en el flujo de bits (correspondiente a 0,1, 2,3 ...). Cuando se utiliza una profundidad de bit superior a 8 se puede aumentar C con  $\log_2(\text{bit depth} - 8)$ .

R1 = 1 y R2 = 0 cuando el corrimiento es positivo, y R1 = 0 y R2 = -1 cuando el corrimiento es negativo.

La indicación de que un corrimiento es positivo o negativo se puede proporcionar en el flujo de bits o ser fija.

Como se ha mencionado anteriormente, todavía se pueden obtener realizaciones adicionales combinando una o más de las realizaciones descritas anteriormente. Ventajosamente, se puede reducir una sobrecarga para codificar corrimientos de borde y banda. Además, la memoria para almacenar los corrimientos de borde y banda puede reducirse utilizando menos corrimientos y con restricción en los valores de corrimiento. Las ventajas con estas realizaciones son el consumo de memoria reducido para almacenar los corrimientos de borde y banda. Además, el decodificador no necesita realizar ningún accionamiento de recorte basado en píxeles después de filtrar con corrimientos de borde o banda.

Los métodos de las realizaciones se realizan en un dispositivo de filtrado que puede estar situado en un codificador o un decodificador como se ilustra esquemáticamente en las figuras 4 y 5. Las figuras 4 y 5 ilustran el ejemplo cuando el método se realiza dentro del bucle de codificación. Es decir, todas las realizaciones del presente documento pueden implementarse en el dispositivo 150 de control de filtrado, designado como 100 en las figuras 4 y 5.

La figura 4 es un diagrama de bloques esquemático de un codificador 40 para codificar un bloque de píxeles en una trama de vídeo de una secuencia de vídeo de acuerdo con una realización.

Se predice un bloque actual de píxeles realizando una estimación de movimiento mediante un estimador 50 de movimiento desde un bloque de píxeles ya proporcionado en la misma trama o en una trama previa. El resultado de la estimación de movimiento es un vector de movimiento o desplazamiento asociado con el bloque de referencia, en el caso de inter-predicción. El vector de movimiento es utilizado por un compensador 50 de movimiento, que está colocalizado con el estimador 50 de movimiento, para emitir una inter-predicción del bloque de píxeles. En otros ejemplos, el compensador de movimiento y el estimador de movimiento pueden ser entidades separadas.

Un intra-predicador 49 calcula una intra-predicción del bloque actual de píxeles. Las salidas del estimador/compensador 50 de movimiento y el intra-predicador 49 se introducen en un selector 51 que selecciona la intra-predicción o la inter-predicción para el bloque actual de píxeles. La salida desde el selector 51 se introduce en un calculador de errores en forma de un sumador 41 que también recibe los valores de píxeles del bloque actual de píxeles. El sumador 41 calcula y emite un error residual como la diferencia de valores de píxeles entre el bloque de píxeles y su predicción.

El error se transforma en un transformador 42, tal como por una transformada de coseno discreta, y se cuantifica por un cuantificador 43 seguido de codificación en un codificador 44, tal como por codificador de entropía. En inter-

codificación, también el vector de movimiento estimado es llevado al codificador 44 para generar la representación codificada del bloque actual de píxeles.

El error residual transformado y cuantificado para el bloque actual de píxeles se proporciona también a un cuantificador inverso 45 y al transformador inverso 46 para recuperar el error residual original. Este error es añadido por un sumador 47 a la salida de predicción de bloque desde el compensador 50 de movimiento o el intra-predicador 49 para crear un bloque de referencia de píxeles que puede utilizarse en la predicción y codificación de un siguiente bloque de píxeles. Este nuevo bloque de referencia es procesado en primer lugar por un dispositivo 100 de control de filtrado, como ejemplo del dispositivo 150 de control de filtrado mencionado anteriormente, con el fin de controlar cualquier filtrado que se aplique al bloque de referencia para combatir cualquier artefacto. El nuevo bloque de referencia procesado se almacena entonces temporalmente en un búfer 48 de trama, donde está disponible para el intra-predicador y el estimador/compensador 50 de movimiento.

La figura 5 es un diagrama de bloques esquemático correspondiente de un decodificador 60 que comprende un dispositivo 100 de control de filtrado de acuerdo con las realizaciones del presente documento. El decodificador 60 comprende una unidad 61 de decodificador, tal como un decodificador de entropía, para decodificar una representación codificada de un bloque de píxeles para obtener un conjunto de errores residuales cuantificados y transformados. Estos errores residuales se descuantifican en un cuantificador inverso 62 y se transforman a la inversa mediante un transformador inverso 63 para obtener un conjunto de errores residuales.

Estos errores residuales se añaden en un sumador 64 a los valores de píxeles de un bloque de referencia de píxeles. El bloque de referencia está determinado por un estimador/compensador 67 de movimiento o un intra-predicador 66, dependiendo de si se realiza inter o intra predicción. En la mayoría de los ejemplos, el estimador/compensador 67 de movimiento sólo realiza la compensación de movimiento. De este modo, un selector 68 está interconectado al sumador 64 y al estimador/compensador 67 de movimiento y al intra-predicador 66. El bloque decodificado de píxeles resultante emitido desde el sumador 64 es introducido en un dispositivo 100 de control de filtrado para controlar cualquier filtro que se aplique para combatir cualquier artefacto. El bloque filtrado de píxeles es emitido desde el decodificador 60 y, además, preferentemente se proporciona temporalmente a un búfer 65 de trama y puede utilizarse como un bloque de referencia de píxeles para un siguiente bloque de píxeles a decodificar. El búfer 65 de trama se conecta de este modo al estimador/compensador 67 de movimiento para hacer disponibles los bloques almacenados de píxeles al estimador/compensador 67 de movimiento.

La salida del sumador 64 se introduce preferentemente también en el intra-predicador 66 para ser utilizado como un bloque de referencia no filtrado de píxeles.

En las realizaciones divulgadas en las figuras 4 y 5, el dispositivo 100 de control de filtrado controla el filtrado en forma del llamado filtrado en bucle. En una implementación alternativa en el decodificador 60, el dispositivo 100 de control de filtrado está dispuesto para realizar el llamado filtrado de post-procesado. En este caso, el dispositivo 100 de control de filtrado funciona sobre las tramas de salida fuera del bucle formado por el sumador 64, el búfer 65 de tramas, el predictor 66, el estimador/compensador 67 de movimiento y el selector 68. Por lo general, no se hace filtrado y control de filtro en el codificador.

La figura 6 es un diagrama de bloques esquemático de un equipo 80 de usuario o terminal de medios que aloja un decodificador 60 con un dispositivo de control de filtrado. El equipo 80 de usuario, como ejemplo del equipo 120 de usuario receptor, puede ser cualquier dispositivo que tenga funciones de decodificación de medios que actúan sobre un flujo de vídeo codificado de tramas de vídeo codificadas para decodificar de este modo las tramas de vídeo y hacer disponibles los datos de vídeo. Ejemplos no limitativos de tales dispositivos incluyen teléfonos móviles y otros reproductores multimedia portátiles, tabletas, ordenador de sobremesa, notebook, grabadoras de vídeo personales, reproductores multimedia, servidores de transmisión de vídeo, receptores de televisión, televisores, ordenadores, decodificadores, consolas de juegos, etc. El equipo 80 de usuario comprende una memoria 84 configurada para almacenar tramas de vídeo codificadas. Estas tramas de vídeo codificadas pueden haber sido generadas por el propio equipo 80 de usuario. Alternativamente, las tramas de vídeo codificadas son generadas por algún otro dispositivo y transmitidas de forma inalámbrica o transmitidas por cable al equipo 80 de usuario. El equipo 80 de usuario comprende entonces un transceptor (transmisor y receptor) o un puerto 82 de entrada y salida para conseguir la transferencia de datos.

Las tramas de vídeo codificadas son llevadas de la memoria 84 a un decodificador 60, tal como el decodificador ilustrado en la figura 5. El decodificador 60 comprende un dispositivo 100 de control de filtrado de acuerdo con las realizaciones. El decodificador 60 decodifica entonces las tramas de vídeo codificadas en tramas de vídeo decodificadas. Las tramas de vídeo decodificadas se proporcionan a un reproductor 86 de medios que está configurado para renderizar las tramas de vídeo decodificadas en datos de vídeo que pueden ser visualizados en un dispositivo de visualización o pantalla 88 del, o conectados al, equipo 80 de usuario.

La figura 6 ilustra otra realización de un equipo 80 de usuario, como un ejemplo de la unidad 130 de envío, que comprende en codificador, tal como el codificador de la figura 4, que comprende un dispositivo de control de filtrado de acuerdo con las realizaciones. El codificador 40 se configura entonces para codificar tramas de vídeo recibidas



por la unidad 82 de I/O y/o generadas por el propio equipo 80 de usuario. En este último caso, el equipo 80 de usuario preferentemente comprende un motor o grabador de medios, tal como en forma de o conectado a una cámara (de vídeo). El equipo 80 de usuario opcionalmente puede comprender también un reproductor 86 de medios, tal como un reproductor 86 de medios con un decodificador y un dispositivo de control de filtrado de acuerdo con las realizaciones, y una pantalla 88.

La figura 7 es un diagrama de bloques esquemático de un equipo 80 de usuario o terminal de medios que aloja un decodificador 60 con un dispositivo de control de filtrado. El equipo 80 de usuario puede ser cualquier dispositivo que tenga funciones de decodificación de medios que funcionan sobre un flujo de vídeo codificado de tramas de video codificadas para decodificar de este modo las tramas de video y hacer disponibles los datos de vídeo. Ejemplos no limitativos de tales dispositivos incluyen teléfonos móviles y otros reproductores multimedia portátiles, tabletas, ordenador de sobremesa, notebook, grabadoras de vídeo personales, reproductores multimedia, servidores de transmisión de vídeo, receptores de televisión, televisores, ordenadores, decodificadores, consolas de juegos, etc. El equipo 80 de usuario comprende una memoria 84 configurada para almacenar tramas de video codificadas. Estas tramas de video codificadas pueden haber sido generadas por el propio equipo 80 de usuario. Alternativamente, las tramas de video codificadas son generadas por algún otro dispositivo y transmitidas de forma inalámbrica o transmitidas por cable al equipo 80 de usuario. El equipo 80 de usuario comprende entonces un transceptor (transmisor y receptor) o un puerto 82 de entrada y salida para conseguir la transferencia de datos.

Las tramas de video codificadas son llevadas de la memoria 84 a un decodificador 60, tal como el decodificador ilustrado en la figura 5. El decodificador 60 comprende un dispositivo 100 de control de filtrado de acuerdo con las realizaciones del presente documento. El decodificador 60 decodifica entonces las tramas de video codificadas en tramas de video decodificadas. Las tramas de video decodificadas se proporcionan a un reproductor 86 de medios que está configurado para renderizar las tramas de video decodificadas en datos de video que pueden visualizarse en un dispositivo de visualización o pantalla 88 o conectados al equipo 80 de usuario.

En la figura 7, se ha ilustrado el equipo 80 de usuario que comprende tanto el decodificador 60 como el reproductor 86 de medios, con el decodificador 60 implementado como parte del reproductor 86 de medios. Sin embargo, esto debe ser visto simplemente como un ejemplo ilustrativo, pero no limitativo, de una realización de implementación para el equipo 80 de usuario. También son posibles implementaciones distribuidas donde el decodificador 60 y el reproductor 86 de medios se proporcionan en dos dispositivos físicamente separados y dentro del alcance del equipo 80 de usuario como se utiliza en el presente documento. La pantalla 88 también podría proporcionarse como un dispositivo separado conectado al equipo 80 de usuario, en el que tiene lugar el procesamiento de datos real.

Por consiguiente, como se ilustra en la figura 8, el dispositivo de control de filtrado implementa las funciones de las realizaciones del presente documento.

Con referencia a la figura 8, se muestra un diagrama de bloques esquemático de un dispositivo 800 de control de filtrado de ejemplo configurado para procesar una primera imagen en un procedimiento de filtrado adaptativo de muestra. El dispositivo de control de filtrado puede ser el dispositivo 150 de control de filtrado de la figura 1. Como se mencionó, la primera imagen se reconstruye a partir de datos de vídeo.

El dispositivo 800 de control de filtrado puede estar comprendido en un dispositivo 110 de red, un decodificador 140 o un codificador 140.

El dispositivo 800 de control de filtrado comprende un circuito 810 de procesamiento configurado para obtener una magnitud de corrimiento de los datos de vídeo. La magnitud de corrimiento indica una magnitud de un corrimiento adaptativo de muestra que se aplicará en el procedimiento de filtrado adaptativo de muestra.

El circuito 810 de procesamiento está configurado además para obtener un signo de corrimiento a partir de los datos de vídeo. El signo de corrimiento indica un signo del corrimiento adaptativo de muestra.

Además, el circuito 810 de procesamiento está configurado para determinar el corrimiento adaptativo de muestra basado en la magnitud de corrimiento y el signo de corrimiento.

Además, el circuito 810 de procesamiento está configurado para procesar, en el procedimiento de filtrado adaptativo de muestra, al menos una porción de la primera imagen mientras se aplica el corrimiento adaptativo de muestra. De esta manera, se puede obtener una segunda imagen.

El circuito 810 de procesamiento puede configurarse adicionalmente para obtener, a partir de los datos de vídeo, una posición de banda que indica una primera banda de un conjunto de bandas consecutivas.

El circuito 810 de procesamiento puede además configurarse para aplicar la magnitud de corrimiento y el signo de corrimiento a una o más de las bandas consecutivas.

El circuito 810 de procesamiento puede además configurarse para obtener un valor a partir de los datos de vídeo,

para mapear el valor a la magnitud de corrimiento de una manera no lineal. El mapeo puede ser cada vez más escaso, en términos de resolución para la magnitud de corrimiento obtenida, mayor será la magnitud de corrimiento.

El circuito 810 de procesamiento puede configurarse adicionalmente para establecer la magnitud de corrimiento basada en la profundidad de bits obtenida a partir de los datos de vídeo para la segunda imagen y para establecer el signo de corrimiento basado en la profundidad de bits, con lo que un valor de pixel de la segunda imagen está entre 0 y un valor máximo de acuerdo con la profundidad de bits.

El circuito 810 de procesamiento puede configurarse adicionalmente para aplicar el corrimiento adaptativo de muestra a una de las bandas consecutivas, en el que el signo de corrimiento es positivo cuando la posición de banda indica una banda inicial entre bandas que cubren valores de píxeles posibles de acuerdo con la profundidad de bits obtenida de los datos de vídeo y el signo de corrimiento es negativo cuando la posición de banda indica una banda para la cual el corrimiento adaptativo de muestra cubre una última banda entre bandas que cubren valores de píxeles posibles de acuerdo con la profundidad de bits obtenida a partir de los datos de vídeo para la segunda imagen.

El circuito 810 de procesamiento puede configurarse adicionalmente para calcular directamente el corrimiento adaptativo de muestra por medio de accionamientos de signo y accionamientos de corrimiento sobre la magnitud de corrimiento y el signo de corrimiento.

Se obtiene una magnitud de cambio del accionamiento de cambio a partir de los datos de vídeo. Adicionalmente o alternativamente, la magnitud de cambio de la operación de cambio está predefinida.

El circuito 810 de procesamiento puede además configurarse para obtener un valor booleano a partir de los datos de vídeo; y derivar la magnitud de corrimiento basada en la profundidad de bits obtenida a partir de los datos de vídeo cuando el valor booleano es verdadero.

El circuito 810 de procesamiento puede ser una unidad de procesamiento, un procesador, un circuito integrado específico de aplicación (ASIC), una matriz de pasarela programable por campo (FPGA) o similar. Como ejemplo, un procesador, un ASIC, una FPGA o similar puede comprender uno o más núcleos de procesador.

El dispositivo 800 de control de filtrado puede comprender además una unidad 820 de determinación configurada para determinar el corrimiento adaptativo de muestra. Por lo tanto, en algunos ejemplos, el circuito 810 de procesamiento puede comprender la unidad 820 de determinación. En otros ejemplos, como se ilustra en la figura 8, la unidad 820 de determinación está separada del circuito 810 de procesamiento.

Adicionalmente, el dispositivo 800 de control de filtrado puede comprender una unidad de obtención configurada para obtener la magnitud de corrimiento, el signo de corrimiento, el valor, la posición de banda y/u otros valores, parámetros o similares descritos en el presente documento.

Además, el dispositivo 800 de control de filtrado puede comprender una unidad de mapeo configurada para mapear el valor a la magnitud de corrimiento.

El dispositivo 800 de control de filtrado comprende además una memoria 830 para almacenar el equipo lógico que se va a ejecutar por, por ejemplo, el circuito de procesamiento. El equipo lógico puede comprender instrucciones para permitir que el circuito de procesamiento realice el método en el dispositivo 150 de control de filtrado como se ha descrito anteriormente junto con la figura 2. La memoria puede ser un disco duro, un medio de almacenamiento magnético, un disquete o disco de ordenador portátil, memoria flash, memoria de acceso aleatorio (RAM) o similar. Además, la memoria puede ser una memoria de registro interna de un procesador.

La figura 9 ilustra esquemáticamente una realización de un ordenador 70 que tiene una unidad 72 de procesamiento, tal como un DSP (procesador de señal digital) o CPU (unidad central de procesamiento). La unidad 72 de procesamiento, como ejemplo del circuito 810 de procesamiento anterior, puede ser una unidad única o una pluralidad de unidades para realizar diferentes pasos del método descrito en el presente documento. El ordenador 70 también comprende una unidad 71 de entrada/salida (I/O) para recibir tramas de vídeo grabadas o generadas o tramas de vídeo codificadas y emitir tramas de vídeo codificadas o datos de vídeo decodificados. La unidad 71 de I/O se ha ilustrado como una unidad única en la figura 9, pero también puede estar en forma de una unidad de entrada separada y una unidad de salida separada.

Además, el ordenador 70 comprende al menos un producto 73 de programa informático en forma de una memoria no volátil, por ejemplo una EEPROM (memoria de sólo lectura programable borrable eléctricamente), una memoria flash o una unidad de disco. El producto de programa informático puede comprender un medio legible por ordenador y un programa informático 74 como se explica a continuación. El programa informático se almacena en el medio legible por ordenador.

Como se ha mencionado, el producto 73 de programa informático comprende el programa informático 74, que

comprende medios de código que cuando se ejecutan o ejecutan por el ordenador 70, tal como por la unidad 72 de procesamiento, hacen que el ordenador 70 realice los pasos del método descrito en lo anterior en relación con la figura 1. Con más detalle, el programa 74 de ordenador es capaz de procesar, en un procedimiento de filtrado adaptativo de muestra, una primera imagen. Como se mencionó, la primera imagen se reconstruye a partir de datos de vídeo. El programa informático 74 comprende unidades de código legibles por ordenador que cuando se ejecutan en el ordenador 70 hacen que el ordenador 70 obtenga una magnitud de corrimiento de los datos de vídeo. La magnitud de corrimiento indica una magnitud de un corrimiento adaptativo de muestra a aplicar en el procedimiento de filtrado adaptativo de muestra y para obtener un signo de corrimiento a partir de los datos de vídeo. El signo de corrimiento indica un signo del corrimiento adaptativo de muestra. Además, se hace que el ordenador determine el corrimiento adaptativo de muestra basado en la magnitud de corrimiento y el signo de corrimiento, y para procesar, en el procedimiento de filtrado adaptativo de muestra, al menos una parte de la primera imagen mientras se aplica el corrimiento adaptativo de muestra. Por lo tanto, en una realización, los medios de código, o las unidades de código, en el programa informático 74 comprenden un módulo 310 que corresponde a cualquier realización descrita en el presente documento. El módulo 310 realiza esencialmente los pasos de los diagramas de flujo en las figuras 2 y 3 cuando se ejecutan en la unidad 72 de procesamiento. De este modo, cuando el módulo 310 se ejecuta en la unidad 72 de procesamiento, corresponden a las correspondientes unidades 810-820 de la figura 8. En tal caso, el equipo 80 de usuario puede comprender adicionalmente o estar conectado a una pantalla para mostrar datos de vídeo.

En la descripción anterior junto con las figuras 8 y 9, la memoria 830, el producto 74 de programa informático y el medio legible por ordenador tienen la misma función o función similar. En algunos ejemplos, una o más de estas entidades pueden combinarse en una entidad. Las realizaciones de la solución se refieren al procesamiento y codificación de imagen y vídeo. Las realizaciones pueden utilizarse en televisión de difusión, televisión por Internet, videoconferencia, almacenamiento de contenido de vídeo/imagen, etc. Las realizaciones pueden ser parte del codificador y/o decodificador en dispositivos móviles así como dispositivos inmóviles, etc.

## Referencias

- [1] C.-M. Fu et al: "Sample adaptive offset for HEVC", 2011 IEEE 13<sup>º</sup> Taller internacional en procesamiento de señal multimedia (MMSP), pp. 1-5, 17 de octubre de 2011.
- [2] T. Yamakage et al: "Description of Core Experiment 8 (CE8): Non-deblocking loop filtering", Equipo colaborativo conjunto en codificación de vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 7<sup>º</sup> Encuentro: Ginebra, CH.
- [3] T.Yamazaki et al: "Improvement of Sample Adaptive Offset with modified bit accuracy and restricted offsets", Equipo colaborativo conjunto en codificación de vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 6<sup>º</sup> Encuentro: Turín, IT, 14-22 de julio de 2011, JCTVC-F396.

## REIVINDICACIONES

- 1.- Un método, realizado por un dispositivo (150) de control de filtrado, para procesar, en un procedimiento de filtrado adaptativo de muestra utilizado en codificación de vídeo de alta eficiencia, HEVC, una primera imagen, en el que la primera imagen es reconstruida a partir de datos de vídeo, comprendiendo el método:
- obtener (201) una magnitud de corrimiento a partir de los datos de vídeo, en el que la magnitud de corrimiento indica una magnitud de un corrimiento adaptativo de muestra `sao_offset`;
  - obtener (204) un signo de corrimiento a partir de los datos de vídeo, en el que el signo de corrimiento indica un signo del corrimiento adaptativo de muestra `sao_offset`,
- en el que el corrimiento adaptativo de muestra `sao_offset` es un corrimiento de banda y/o un corrimiento de borde;
- en el que el corrimiento adaptativo de muestra `sao_offset` es codificado por código de longitud variable o código de longitud fija o codificación aritmética binaria adaptativa por contexto, CABAC, y el signo de corrimiento y la magnitud de corrimiento se codifican de forma separada en los datos de vídeo;
- en el que un mapeo (203) existe entre el corrimiento adaptativo de muestra `sao_offset` y un valor adicional de corrimiento adaptativo de muestra `SaoOffsetVal`, siendo el mapeo:
- $$\text{sao\_offset} = \log_2(\text{abs}(\text{SaoOffsetVal})) * \text{sign}(\text{SaoOffsetVal})$$
- determinar (205) el valor adicional de corrimiento adaptativo de muestra `SaoOffsetVal` basándose en la magnitud de corrimiento y el signo de corrimiento; y
  - procesar (207), en el procedimiento de filtrado adaptativo de muestra, al menos una porción de la primera imagen aplicando el valor adicional de corrimiento adaptativo de muestra `SaoOffsetVal`.
- 2.- El método de acuerdo con la reivindicación 1, que comprende además:
- obtener (206), de los datos de vídeo, una posición de banda que indica una banda primera e un conjunto de bandas consecutivas, en el que la posición de banda determina una posición de la banda primera a la que el valor adicional de corrimiento adaptativo de muestra `SaoOffsetVal` debería aplicarse.
- 3.- El método de acuerdo con la reivindicación 2, en el que la magnitud de corrimiento y el signo de corrimiento se aplican a uno o más de los conjuntos de bandas consecutivas.
- 4.- El método de acuerdo con cualquiera de las reivindicaciones precedentes, en el que la determinación (205) del valor adicional de corrimiento adaptativo de muestra `SaoOffsetVal` comprende directamente calcular el valor adicional de corrimiento adaptativo por medio de accionamientos de signo y accionamientos de cambio sobre la magnitud de corrimiento y el signo de corrimiento, en el que la magnitud de corrimiento se codifica con una base dos.
- 5.- El método de acuerdo con la reivindicación 4, en el que una magnitud de cambio del accionamiento de cambio se obtiene de los datos de vídeo, o en el que una magnitud de cambio del accionamiento de cambio es predefinida.
- 6.- Un dispositivo (150) de control de filtrado, configurado para procesar, en un procedimiento de filtrado adaptativo de muestra utilizado en codificación de vídeo de alta eficiencia, HEVC, una primera imagen, en el que la primera imagen es reconstruida a partir de datos de vídeo, comprendiendo el dispositivo (150) de control de filtrado un circuito (810) de procesamiento configurado para:
- obtener una magnitud de corrimiento a partir de los datos de vídeo, en el que la magnitud de corrimiento indica una magnitud de un corrimiento adaptativo de muestra `sao_offset`;
  - obtener un signo de corrimiento a partir de los datos de vídeo, en el que el signo de corrimiento indica un signo del corrimiento adaptativo de muestra `sao_offset`,
- en el que el corrimiento adaptativo de muestra `sao_offset` es un corrimiento de banda y/o un corrimiento de borde;
- en el que el corrimiento adaptativo de muestra `sao_offset` es codificado por código de longitud variable o código de longitud fija o codificación aritmética binaria adaptativa por contexto, CABAC y el signo de corrimiento y la magnitud de corrimiento están codificados de forma separada en los datos de vídeo;
- en el que un mapeo (203) existe entre el corrimiento adaptativo de muestra `sao_offset` y un valor adicional de

corrimiento adaptativo de muestra SaoOffsetVal, siendo el mapeo:

$$\text{sao\_offset} = \log_2(\text{abs}(\text{SaoOffsetVal})) * \text{sign}(\text{SaoOffsetVal})$$

- 5 - determinar (205) el valor adicional de corrimiento adaptativo de muestra SaoOffsetVal basándose en la magnitud de corrimiento y el signo de corrimiento; y
- procesar (207), en el procedimiento de filtrado adaptativo de muestra, al menos una porción de la primera imagen aplicando el valor adicional de corrimiento adaptativo de muestra SaoOffsetVal.
- 10 7.- El dispositivo (150) de control de filtrado de acuerdo con la reivindicación 6, en el que el circuito (810) de procesamiento se configura además para obtener, de los datos de vídeo, una posición de banda que indica una banda primera de un conjunto de bandas consecutivas, en el que la posición de banda determina una posición de la banda primera a la que el valor adicional de corrimiento adaptativo de muestra SaoOffsetVal debería aplicarse.
- 15 8.- El dispositivo (150) de control de filtrado de acuerdo con la reivindicación 7, en el que el circuito (810) de procesamiento se configura además para aplicar la magnitud de corrimiento y el signo de corrimiento a una o más de las bandas consecutivas.
- 20 9.- El dispositivo (150) de control de filtrado de acuerdo con cualquiera de las reivindicaciones 6-8, en el que el circuito (810) de procesamiento se configura además para calcular el valor adicional de corrimiento adaptativo de muestra SaoOffsetVal calculando directamente el valor adicional de corrimiento adaptativo de muestra SaoOffsetVal por medio de accionamientos de signo y accionamientos de cambio sobre la magnitud de corrimiento y el signo de corrimiento, en el que la magnitud de corrimiento es codificado con una base dos.
- 25 10.- El dispositivo (150) de control de filtrado de acuerdo con la reivindicación 9, en el que una magnitud de cambio del accionamiento de cambio se obtiene a partir de los datos de vídeo, o en el que una magnitud del accionamiento es predefinida.
- 30 11.- El dispositivo (150) de control de filtrado de acuerdo con cualquiera de las reivindicaciones 6-10, estando comprendido en un dispositivo (110) de red, un decodificador (140) o un codificador (140).
- 12.- Un programa informático (74) para procesar, en un procedimiento de filtrado adaptativo de muestra, una primera imagen, en el que la primera imagen es reconstruida a partir de datos de vídeo, en el que el programa informático (74) comprende unidades de código legibles por ordenador que cuando se ejecutan en un ordenador (70) hacen que el ordenador (70) realice un método de acuerdo con cualquiera de las reivindicaciones 1-5.
- 35 13.- Un producto (73) de programa informático, que comprende un medio legible por ordenador y un programa informático (74) de acuerdo con la reivindicación 12 almacenado en el medio legible por ordenador.

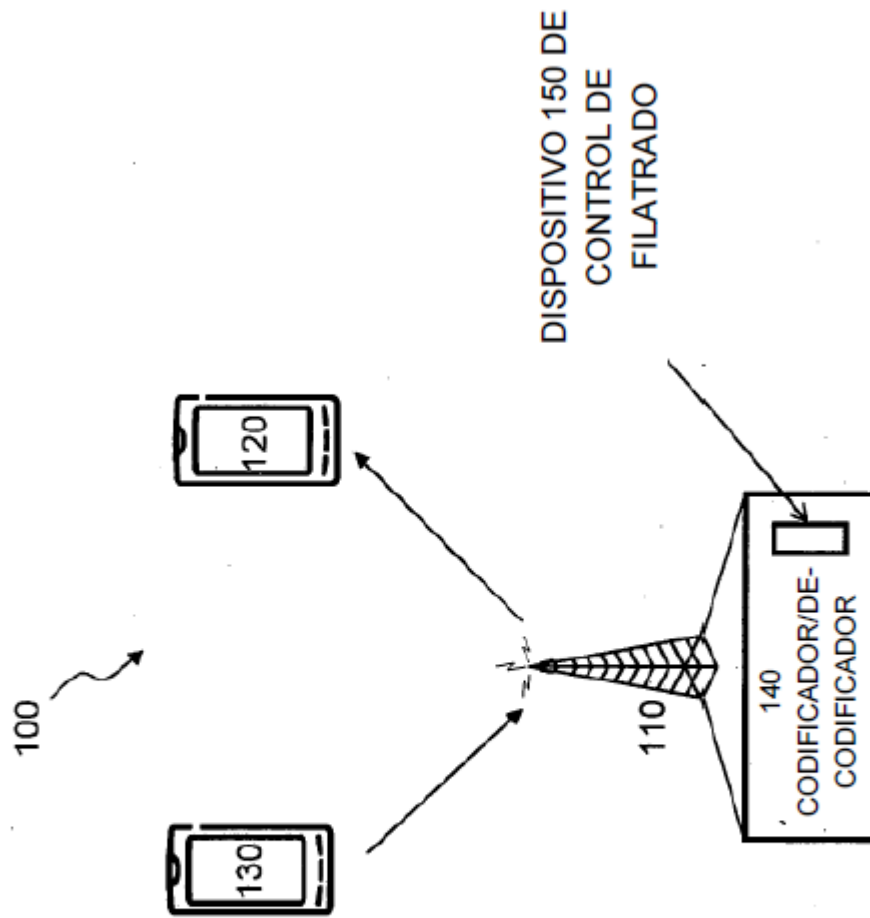


FIG.1

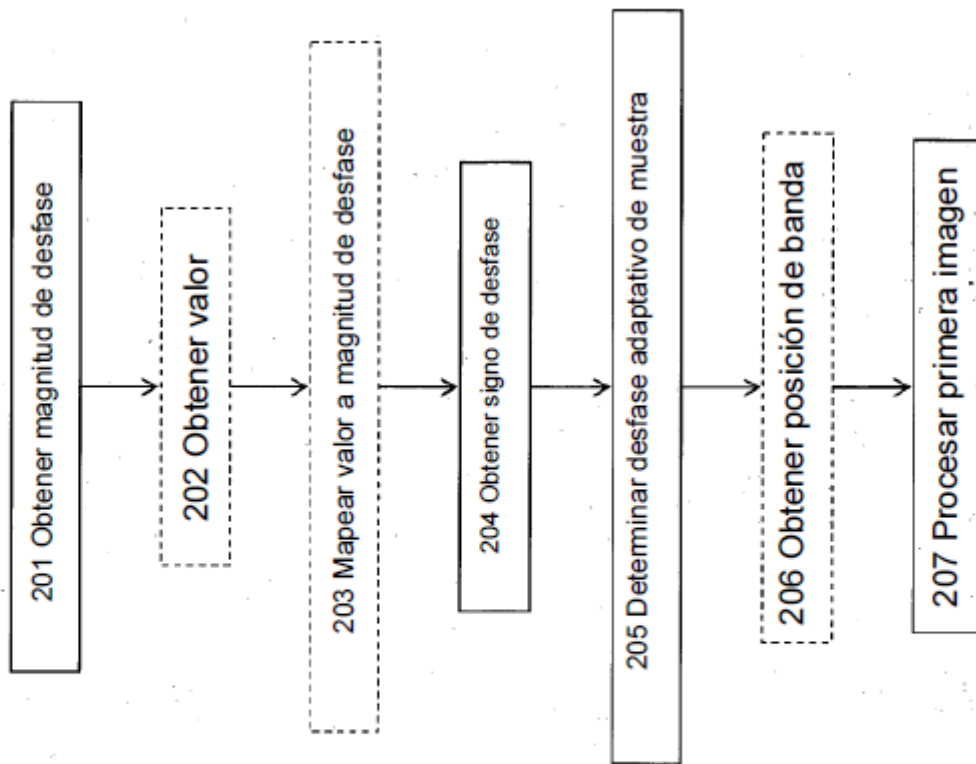


FIG. 2

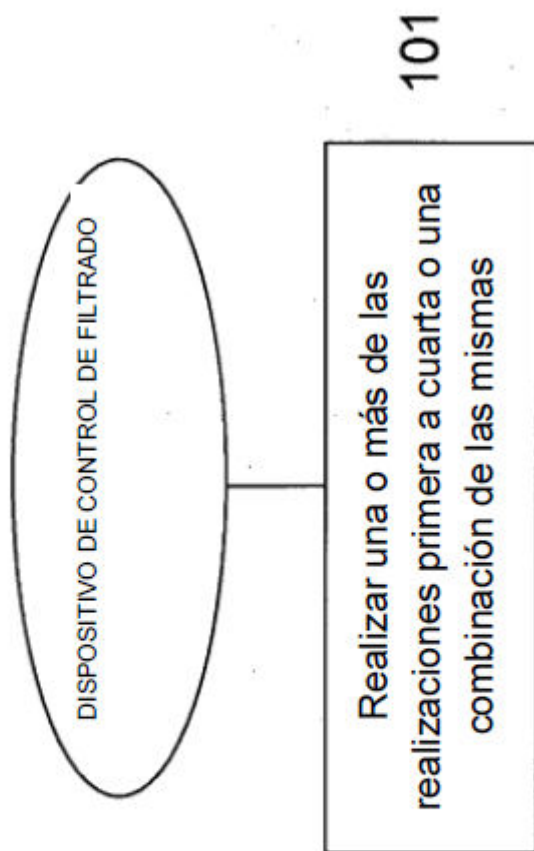


FIG. 3



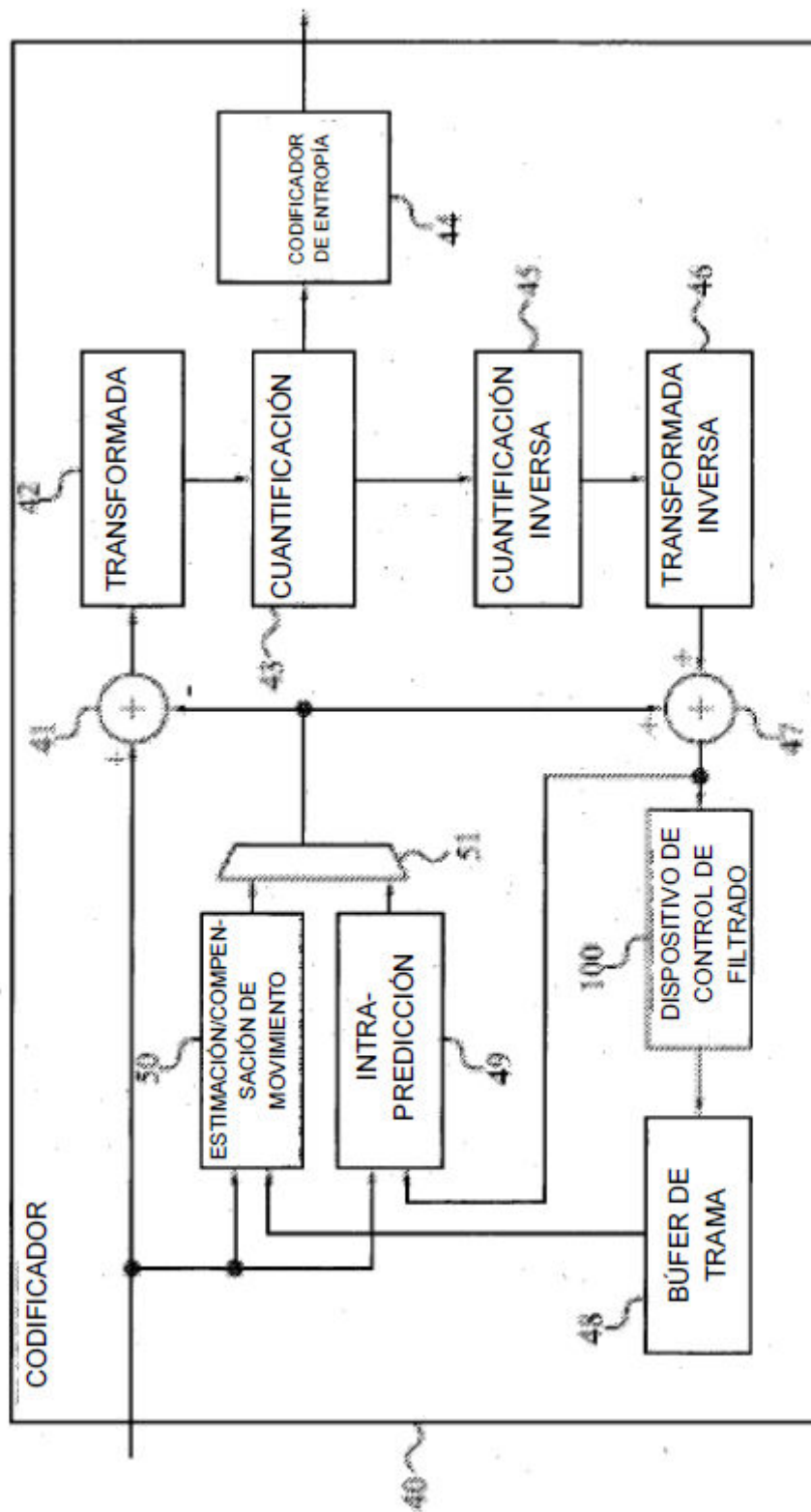


FIG.4

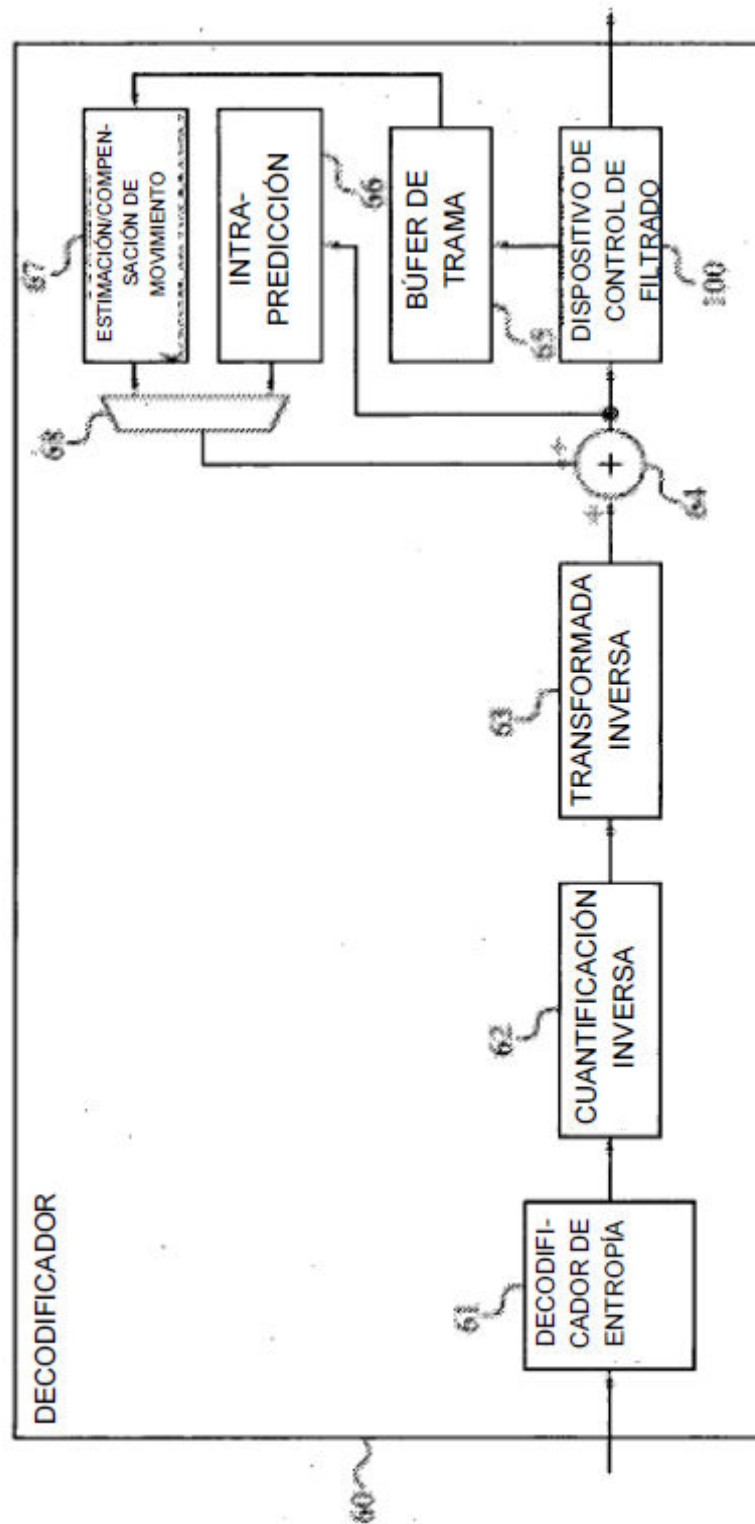


FIG.5

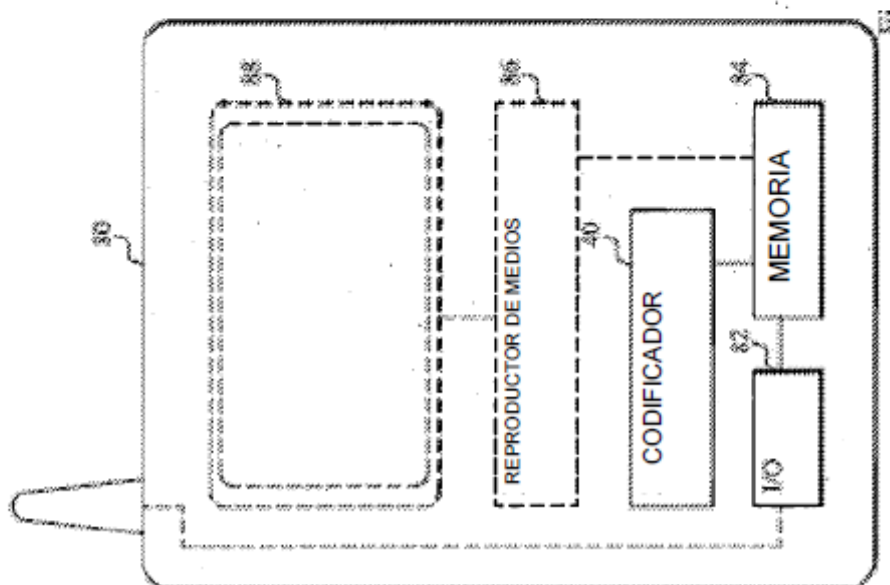


FIG. 6

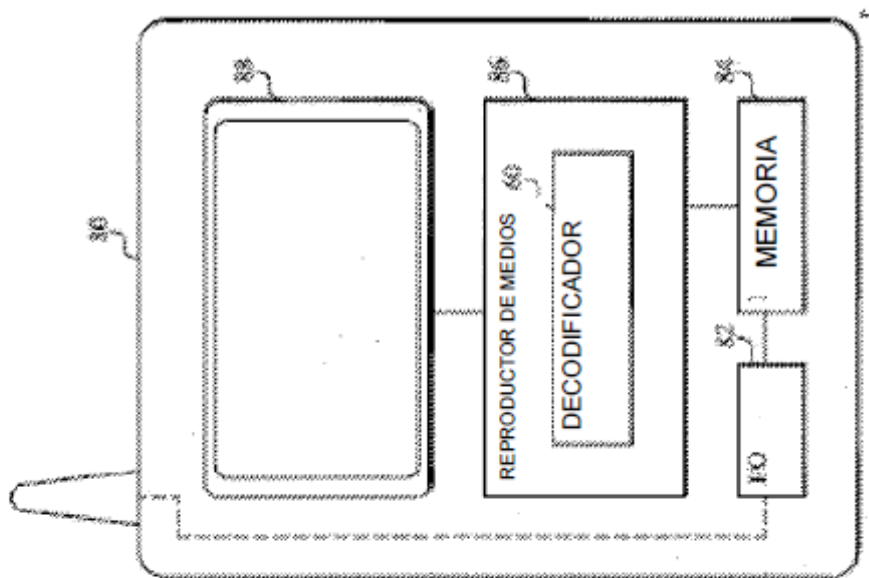


FIG. 7

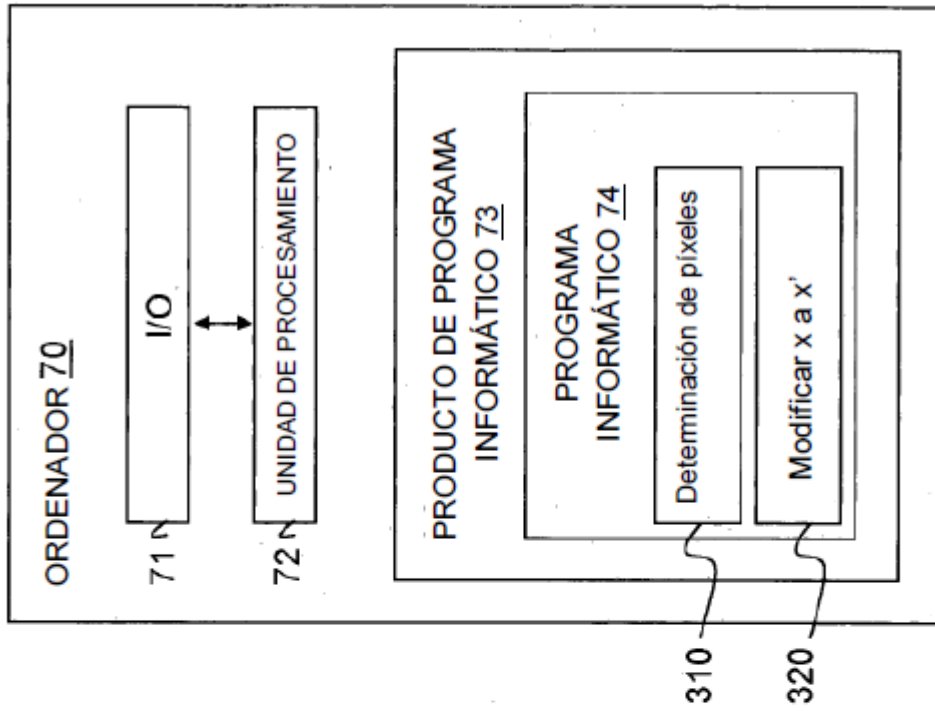


FIG. 9

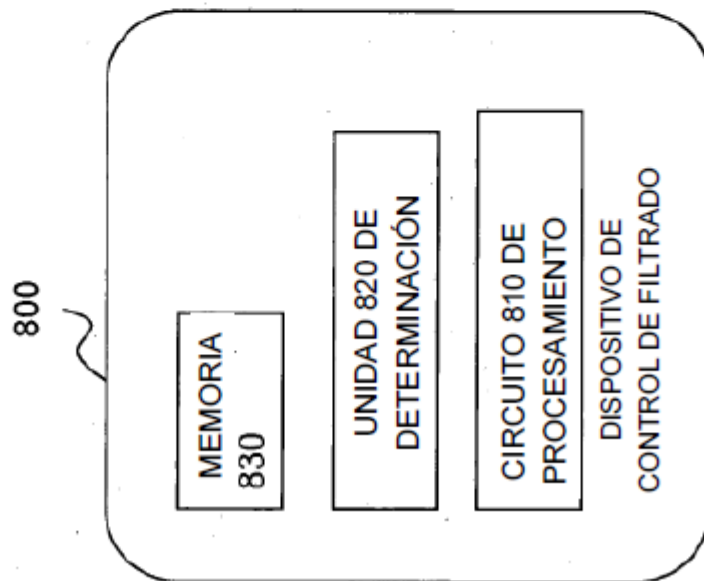


FIG. 8