

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 548 043**

51 Int. Cl.:

H04N 19/176 (2014.01)

H04N 19/61 (2014.01)

H04N 19/117 (2014.01)

H04N 19/14 (2014.01)

H04N 19/82 (2014.01)

H04N 19/86 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **06.10.2011 E 11855586 (1)**

97 Fecha y número de publicación de la concesión europea: **12.08.2015 EP 2664141**

54 Título: **Filtrado de desbloqueo**

30 Prioridad:

14.01.2011 US 201161432751 P

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

13.10.2015

73 Titular/es:

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

72 Inventor/es:

**NORKIN, ANDREY;
ANDERSSON, KENNETH y
SJÖBERG, RICKARD**

74 Agente/Representante:

DE ELZABURU MÁRQUEZ, Alberto

ES 2 548 043 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 de desbloqueo

Sector técnico

5 La presente descripción se refiere al filtrado de desbloqueo para la reducción de aberraciones de bloqueo en las fronteras de bloque.

Antecedentes

10 Los filtros de desbloqueo se utilizan en los estándares de codificación de video para combatir las aberraciones de bloqueo. Las aberraciones de bloqueo se producen debido a que el video original se divide en bloques que son procesados con relativa independencia. Las aberraciones de bloqueo pueden, por ejemplo, producirse debido a diferentes intra predicciones de los bloques, a efectos de cuantificación y a compensación de movimiento. A continuación se describen dos variantes particulares de desbloqueo.

Desbloqueo H.264

15 En la codificación de video más avanzada, tal como la H.264, existe un filtro de desbloqueo, denominado también filtro de bucle, tras la predicción y la reconstrucción residual, pero antes del almacenamiento de la reconstrucción para posterior referencia cuando se codifican o descodifican los fotogramas siguientes. El filtrado de desbloqueo consiste en varias etapas tales como decisiones de filtro, operaciones de filtrado, una función de acortamiento y cambios de valores de píxel. La decisión de filtrar o no la frontera se toma basándose en la evaluación de varias condiciones. Las decisiones de filtro dependen del tipo de macrobloque (MB), de la diferencia del vector de movimiento (MV) entre bloques vecinos, si los bloques vecinos tienen residuos codificados y de la estructura local de los bloques actual y/o vecino.

20 Entonces, la cantidad de filtrado para un píxel depende, entre otras cosas, de la posición de ese píxel con respecto a la frontera del bloque y al valor del parámetro de cuantificación (QP – Quantization Parameter, en inglés) utilizado para codificación residual.

25 La decisión de filtro se basa en la comparación de tres diferencias de píxel con tres umbrales. Los umbrales están adaptados al parámetro de cuantificación (QP). Por ejemplo, asúmase una frontera de bloque vertical de

$$a \ b \ c \ d \ | \ e \ f \ g \ h$$

30 en la que a, b, c y d denotan valores de píxel de una línea de píxeles en el bloque actual, denotando e, f, g y h los correspondientes valores de píxel de los píxeles de una fila de píxeles correspondiente en el bloque vecino. Si se cumplen las siguientes condiciones la decisión de filtro es positiva, por ejemplo $\text{abs}(d - e) < \text{thr1}$, $\text{abs}(c - d) < \text{thr2}$ y $\text{abs}(e - f) < \text{thr2}$, donde thr1 y thr2 están adaptados sobre la base del QP.

35 Existen dos modos de filtrado en H.264. En el primer modo de filtrado, denominado filtrado normal, el filtrado puede describirse con un valor delta con el cual el filtrado intercambia el valor actual. El filtrado para los píxeles más cercanos a la frontera de bloque es $d' = d + \text{delta}$ y $e' = e - \text{delta}$, donde delta ha sido acortado hasta un umbral $\pm \text{thr3}$, hasta un valor que está restringido por el QP. Se permite por ello más filtrado para un QP alto que para un QP bajo. El acortamiento puede describirse como desfase acortado = $\max(-\text{thr3}, \min(\text{thr3}, \text{delta}))$, donde thr3 controla la potencia del filtro. Un valor mayor que thr3 significa que el filtrado es más fuerte, lo que significa que se producirá un efecto de filtrado de paso bajo más fuerte.

40 La potencia del filtro puede aumentar si se cumple también cualquiera de las dos condiciones siguientes, por ejemplo $\text{abs}(b - d) < \text{thr2}$ y $\text{abs}(e - g) < \text{thr2}$. La potencia del filtro se adapta al acortar menos el delta, por ejemplo permitir una mayor variación.

El segundo modo de filtrado, denominado filtrado fuerte, se aplica solo para fronteras intra macrobloques, cuando se cumple la condición siguiente: $\text{abs}(d - e) < \text{thr1} / 4$.

Para más información sobre el filtrado de desbloqueo en H.264, se hace referencia a List et al., Adaptive Deblocking Filter, IEEE Transactions on Circuits and Systems for Video Technology, vol. 13, nº 7, Julio de 2003.

45 Desbloqueo en el borrador de HEVC

50 En el borrador de la especificación de HECV (Codificación de Video de Alta Eficiencia) "Test Model under Consideration", documento ITU-T SG16 WP3, JCTVC-B205, Capítulo 6.5, Proceso de filtro en bucle, el filtro de desbloqueo funciona de manera diferente con respecto a H.264. El filtrado se lleva a cabo si al menos uno de los bloques en el lado de la frontera es intra, o tiene coeficientes distintos de cero, o la diferencia entre las componentes del vector de movimiento de los bloques es mayor o igual a un píxel entero. Por ejemplo, cuando se filtra la frontera entre los bloques con una frontera de bloque vertical de

$$p_3, p_2, p_1, p_0 \mid q_0, q_1, q_2, q_3,$$

siendo p_j el valor de píxel del número de píxel j de número de fila i en el bloque actual y siendo q_j el valor de píxel del número de píxel j de número de fila i en el bloque vecino, $i = 0 \dots 7$, $j = 0 \dots 3$, entonces debería satisfacerse también la condición siguiente:

$$5 \quad d = |p_2 - 2 \times p_1 + p_0| + |q_2 - 2 \times q_1 + q_0| + |p_5 - 2 \times p_4 + p_3| + |q_5 - 2 \times q_4 + q_3| < \beta$$

donde β depende de QP. En la especificación HEVC mencionada anteriormente, existe una tabla de β , donde β aumenta con QP.

Si se cumplen las condiciones y se realiza el filtrado entre el bloque actual y el bloque vecino, se lleva a cabo uno de los dos tipos de filtrado, denominados filtrado débil y filtrado fuerte, respectivamente. La elección entre el filtrado fuerte y el débil se realiza de manera separada para cada línea dependiendo de las condiciones siguientes. Para cada línea $i = 0 \dots 7$, el filtrado fuerte se realiza si todas las condiciones siguientes son verdaderas; de lo contrario, se realiza un filtrado débil:

$$\begin{aligned} d &< (\beta \gg 2) \\ (|p_3 - p_0| + |q_0 - q_3|) &< (\beta \gg 3) \\ |p_0 - q_0| &< ((5 \times t_c + 1) \gg 1) \end{aligned}$$

donde t_c y β dependen de QP y \gg denota un operador de desplazamiento hacia la derecha.

15 *Filtrado débil en el borrador de HEVC*

El filtrado débil se realiza basándose en las condiciones anteriores. El filtrado real funciona calculando un desfase (Δ), sumándolo al valor original del píxel y acortando (Clip, en inglés) la suma hasta un valor de píxel de resultado filtrado en el rango de 0 – 255:

$$\begin{aligned} \Delta &= \text{Clip}(-t_c, t_c, (13 \times (q_0 - p_0) + 4 \times (q_1 - p_1) - 5 \times (q_2 - p_2) + 16) \gg 5) \\ p_0 &= \text{Clip}_{0-255}(p_0 + \Delta) \\ q_0 &= \text{Clip}_{0-255}(q_0 - \Delta) \\ p_1 &= \text{Clip}_{0-255}(p_1 + \Delta/2) \\ q_1 &= \text{Clip}_{0-255}(q_1 - \Delta/2) \end{aligned}$$

20 donde la función de acortado Clip (A, B, x) se define como Clip (A, B, x) = A si $x < A$, Clip (A, B, x) = B si $x > B$, Clip (A, B, x) = x si $A \leq x \leq B$ y Clip₀₋₂₅₅(x) se define como Clip (0, 255, x).

Filtrado fuerte en el borrador de HEVC

El modo de filtrado fuerte se lleva a cabo mediante el conjunto de operaciones siguiente:

$$\begin{aligned} p_0 &= \text{Clip}_{0-255}((p_2 + 2 \times p_1 + 2 \times p_0 + 2 \times q_0 + q_1 + 4) \gg 3) \\ q_0 &= \text{Clip}_{0-255}((p_1 + 2 \times p_0 + 2 \times q_0 + 2 \times q_1 + q_2 + 4) \gg 3) \\ p_1 &= \text{Clip}_{0-255}((p_2 + p_1 + p_0 + q_0 + 2) \gg 2) \\ q_1 &= \text{Clip}_{0-255}((p_0 + q_0 + q_1 + q_2 + 2) \gg 2) \\ p_2 &= \text{Clip}_{0-255}((2 \times p_3 + 3 \times p_2 + p_1 + p_0 + q_0 + 4) \gg 3) \\ q_2 &= \text{Clip}_{0-255}((p_0 + q_0 + q_1 + 3 \times q_2 + 2 \times q_3 + 4) \gg 3) \end{aligned}$$

25 Un problema con el desbloqueo en HEVC es que el filtro débil no tiene buenas características en paso bajo. Puede verse que los píxeles primero y segundo de la frontera de bloque se modifican sumando o restando Δ y $\Delta/2$ respectivamente. Tal filtro no puede eliminar las altas frecuencias que pueden aparecer cerca de la frontera de bloque y no puede eliminar el zumbido. Además, las características de frecuencia del filtro en HEVC muestran ligeras amplificaciones de las frecuencias mayores cuando se filtra el segundo píxel (p_1 y q_1) de la frontera de bloque, lo que puede resultar en un peor comportamiento de la distorsión de velocidad (RD – Rate Distortion, en inglés) del codificador. Además, el desfase para el píxel p_1 es igual en magnitud al desfase del píxel q_1 aunque la estructura de la señal pueda ser diferente en lados diferentes de la frontera de bloque. Por lo tanto, el filtro HEVC no se adapta bien a la estructura local del lado de la frontera de bloque.

Por ello, existe la necesidad de un filtro de desbloqueo eficiente que pueda ser utilizado para la reducción de las aberraciones de bloqueo en las fronteras de bloque y que no tenga los inconvenientes mencionados anteriormente.

Compendio

5 Un objetivo general es combatir las aberraciones de bloqueo en las fronteras de bloque entre bloques de píxeles en un fotograma de video.

Las realizaciones descritas en esta memoria alcanzan estos y otros objetivos.

Un aspecto de las realizaciones define un método para la reducción de las aberraciones de bloqueo en una frontera de bloque entre un bloque de píxeles y un bloque vecino de píxeles en un fotograma de video. El método calcula un primer desfase para que sea igual a $(9 \times (q_0 - p_0) - 3 \times (q_1 - p_1) + 8) \gg 4$, donde \gg denota una operación de desplazamiento hacia la derecha, p_0 denota un valor de píxel de un píxel más cercano a la frontera de bloque en una línea de píxeles en el bloque, p_1 denota un valor de píxel de un píxel siguiente más cercano a la frontera de bloque en la línea de píxeles, q_0 denota un valor de píxel de un píxel más cercano al frontera de bloque en una línea de píxeles correspondiente en el bloque vecino y q_1 denota un valor de píxel de un píxel siguiente más cercano a la frontera de bloque en la línea de píxeles correspondiente. La línea de píxeles y la línea de píxeles correspondiente son perpendiculares a la frontera de bloque. El valor de píxel del píxel más cercano a la frontera de bloque en la línea de píxeles es modificado sumando el primer desfase al valor de píxel de este píxel para formar un valor de píxel modificado. Correspondientemente, el valor de píxel del píxel más cercano a la frontera de bloque pero en la línea de píxeles correspondiente es modificado restando el primer desfase del valor de píxel de este píxel para formar un valor de píxel modificado. El método comprende además el acortamiento del primer desfase para que esté dentro de un intervalo de $-t_c$ y t_c ajustando el primer desfase para que sea igual a $-t_c$ si el primer desfase es menor que $-t_c$, y configurando el primer desfase para que sea igual a t_c si el primer desfase es mayor que t_c , donde t_c es un valor de umbral que depende de un valor del parámetro de cuantificación asignado al bloque.

25 En una realización opcional de este aspecto, el método comprende además el acortamiento del valor de píxel modificado del píxel más cercano a la citada frontera de bloque en la línea de píxeles para que esté dentro de un intervalo de cero y un valor máximo definido, y el acortamiento del valor de píxel modificado del píxel más cercano a la frontera de bloque en la línea de píxeles correspondiente para que esté dentro de un intervalo de cero y un valor máximo definido. Este acortamiento opcional se consigue ajustando el valor de píxel modificado para que sea igual a cero si el valor de píxel modificado es menor que cero, y ajustando el valor de píxel modificado para que sea igual al valor máximo definido si el valor de píxel modificado es mayor que el valor máximo definido.

30 En una realización opcional de este aspecto, el método comprende además el cálculo de un segundo desfase que estará basado en $(p_2 + p_0 - 2 \times p_1 + 2 \times \Delta) / 4$, donde p_2 denota un valor de píxel de un segundo píxel siguiente más cercano a la frontera de bloque en la línea de píxeles y Δ denota el primer desfase. El método comprende también, en esta realización opcional, el cálculo de un tercer desfase que estará basado en $(q_2 + q_0 - 2 \times q_1 - 2 \times \Delta) / 4$, en el que q_2 denota un valor de píxel de un segundo píxel siguiente más cercano a la frontera de bloque en la línea de píxeles correspondiente. El valor de píxel del píxel siguiente más cercano a la frontera de bloque en la línea de píxeles es entonces modificado sumando el segundo desfase al valor de píxel siguiente al más cercano a la frontera de bloque en la línea de píxeles para formar un valor de píxel modificado. Correspondientemente, el método comprende también, en esta realización opcional, la modificación del valor de píxel del píxel siguiente más cercano a la frontera de bloque, en la línea de píxeles correspondiente sumando el tercer desfase al valor de píxel del píxel siguiente al más cercano a la frontera de bloque en la línea de píxeles correspondiente para formar un valor de píxel modificado.

En una realización opcional de este aspecto, el segundo desfase se calcula para que sea igual a $((p_2 + p_0 + 1) \gg 1) - p_1 + \Delta \gg 1$, donde \gg denota una operación de desplazamiento hacia la derecha y el tercer desfase se calcula para que sea igual a $((q_2 + q_0 + 1) \gg 1) - q_1 - \Delta \gg 1$.

45 En una realización opcional de este aspecto, el método comprende además el acortamiento del segundo desfase dentro de un intervalo $-t_{c2}$ y t_{c2} , y el acortamiento del tercer desfase para que esté dentro del intervalo de $-t_{c2}$ y t_{c2} , y donde t_{c2} es un valor de umbral que depende de un valor del parámetro de cuantificación asignado al bloque. Este acortamiento opcional se consigue configurando el desfase segundo o tercero para que sea igual a $-t_{c2}$ si el desfase segundo o tercero es menor que $-t_{c2}$ y configurando el desfase segundo o tercero para que sea igual a t_{c2} si el desfase segundo o tercero es mayor que t_{c2} .

En una realización opcional de este aspecto, el método comprende además el acortamiento del valor de píxel modificado del píxel siguiente más cercano a la frontera de bloque en la línea de píxeles para que esté dentro de un intervalo de cero a un valor máximo definido, y el acortamiento del valor de píxel modificado del siguiente píxel más cercano a la frontera de bloque en la línea de píxeles correspondiente para que esté dentro de un intervalo de cero y

el valor máximo definido. Este acortamiento opcional se consigue ajustando el valor de píxel modificado para que sea igual a cero si el valor de píxel modificado es menor que cero y ajustando el valor de píxel modificado para que sea igual al valor máximo definido si el valor de píxel modificado es mayor que el valor máximo definido.

En otra realización de este aspecto, el método comprende además el cálculo de un cuarto desfase para que esté

5 basado en $(p_3 + p_1 - 2 \times p_2 + 2 \times \Delta_{p1})/4$, donde p_3 denota un valor de píxel de un píxel tercero siguiente más cercano a la frontera de bloque en la línea de píxeles y Δ_{p1} denota el segundo desfase. El método comprende también, en esta realización opcional, el cálculo de un quinto desfase para que esté basado en

10 $(q_3 + q_1 - 2 \times q_2 + 2 \times \Delta_{q1})/4$, donde q_3 denota un valor de píxel de un tercer píxel más cercano a la frontera de bloque en la línea de píxeles correspondiente y Δ_{q1} denota el tercer desfase. El valor de píxel del segundo píxel siguiente más cercano a la frontera de bloque en la línea de píxeles se modifica sumando el cuarto desfase al valor de píxel del segundo píxel más cercano a la frontera de bloque en la línea de píxeles para formar un valor de píxel modificado. El método comprende también, en esta realización opcional, la modificación del valor de píxel del segundo píxel siguiente más cercano a la frontera de bloque en la línea de píxeles correspondiente sumando el quinto desfase al valor de píxel del segundo píxel siguiente más cercano a la frontera de bloque en la línea de píxeles correspondiente para formar un valor de píxel modificado.

En una realización opcional de este aspecto, el cuarto desfase se calcula para que sea igual a

15 $((p_3 + p_1 + 1) \gg 1) - p_2 + \Delta_{p1} \gg 1$, donde \gg denota una operación de desplazamiento hacia la derecha, y el quinto desfase se calcula para que sea igual a $((q_3 + q_1 + 1) \gg 1) - q_2 + \Delta_{q1} \gg 1$.

En una realización opcional de este aspecto, una unidad de filtro de desbloqueo calcula el primer desfase para que

20 esté basado en $(9 \times (q_0 - p_0) - 3 \times (q_1 - p_1))/16$. La unidad de filtro de desbloqueo modifica el valor de píxel del píxel más cercano a la frontera de bloque en la línea de píxeles sumando el primer desfase al valor de píxel del píxel más cercano a la frontera de bloque en la línea de píxeles para formar un valor de píxel modificado. La unidad de filtro de desbloqueo modifica también el valor de píxel del píxel más cercano a la frontera de bloque en la línea de píxeles correspondiente restando el primer desfase del valor de píxel del píxel más cercano a la frontera de bloque en la línea de píxeles correspondiente para formar un valor de píxel modificado.

Otro aspecto de las realizaciones define una unidad de filtro de desbloqueo para reducir las aberraciones de bloqueo en una frontera de bloque entre un bloque de píxeles y un bloque de píxeles vecino en un fotograma de video. La unidad de filtro de desbloqueo comprende un calculador de primer desfase configurado para calcular un primer

30 desfase para que sea igual a $(9 \times (q_0 - p_0) - 3 \times (q_1 - p_1) + 8) \gg 4$, donde \gg denota una operación de desplazamiento a la derecha, p_0 denota un valor de píxel más cercano a la frontera de bloque en una línea de píxeles en el bloque, p_1 denota un valor de píxel de un píxel siguiente más cercano a la frontera de bloque en la línea de píxeles, q_0 denota un valor de píxel de un píxel más cercano a la frontera de bloque en una línea de píxeles correspondiente en el bloque vecino y q_1 denota un valor de píxel de un píxel siguiente más cercano a la frontera de bloque en la línea de píxeles correspondiente. La línea de píxeles y la línea de píxeles correspondiente son perpendiculares a la frontera de bloque. La unidad de filtro de desbloqueo comprende también un valor de píxel modificado configurado para modificar el valor de píxel del píxel más cercano a la frontera de bloque en la línea de píxeles sumando el primer desfase al valor de píxel de este píxel para formar un valor de píxel modificado. El valor de píxel modificado está configurado también para modificar el valor de píxel del píxel más cercano a la frontera de bloque pero en la línea de píxeles correspondiente, restando el primer desfase del valor de píxel de este píxel para formar un valor de píxel modificado. La unidad de filtro de desbloqueo comprende también una unidad de primer acortamiento configurada para acortar el primer desfase dentro del intervalo de $-t_c$ y t_c , ajustando el primer desfase para que sea igual a $-t_c$ si el primer desfase es menor que $-t_c$ y ajustando el primer desfase para que sea igual a t_c si el primer desfase es mayor que t_c , donde t_c es un valor de umbral que depende del valor de un parámetro de cuantificación asignado al bloque.

45 En una realización opcional de este aspecto, la unidad de filtro de desbloqueo comprende una unidad de segundo acortamiento configurada para acortar el valor de píxel modificado más cercano a la frontera de bloque en la línea de píxeles para que esté dentro de un intervalo de cero a un valor máximo definido, y acortar el valor de píxel modificado del píxel más cercano a la frontera de bloque en la línea de píxeles correspondiente para que esté dentro de un intervalo de cero y el valor máximo definido. Este acortamiento opcional por parte de la unidad de segundo acortamiento se consigue ajustando el valor de píxel modificado para que sea igual a cero si el valor de píxel modificado es menor que cero y ajustando el valor de píxel modificado para que sea igual al número máximo definido si el valor de píxel modificado es mayor que el valor máximo definido.

En una realización opcional de este aspecto, la unidad de filtro de desbloqueo comprende un calculador de segundo

desfase configurado para calcular un segundo desfase que esté basado en $(p_2 + p_0 - 2 \times p_1 + 2 \times \Delta) / 4$, donde p_2 denota un valor de píxel de un píxel segundo siguiente más cercano a la frontera de bloque en la línea de píxeles y Δ denota el primer desfase. La unidad de filtro de desbloqueo comprende también, en esta realización opcional, un

5 calculador de tercer desfase configurado para calcular un tercer desfase que se base en $(q_2 + q_0 - 2 \times q_1 - 2 \times \Delta) / 4$, donde q_2 denota un valor de píxel de un píxel segundo siguiente más cercano a la frontera de bloque en la línea de píxeles correspondiente. En esta realización opcional el modificador del valor de píxel está configurado para modificar el valor de píxel del píxel siguiente más cercano a la frontera de bloque en la línea de píxeles, sumando el segundo desfase al valor de píxel del píxel siguiente más cercano a la frontera de bloque en la línea de píxeles para formar un valor de píxel modificado, y para modificar el valor de píxel del píxel siguiente más cercano a la frontera de bloque en la línea de píxeles correspondiente sumando el tercer desfase al valor de píxel del píxel siguiente más cercano a la frontera de bloque en la línea de píxeles correspondiente para formar un valor de píxel modificado.

En una realización opcional de este aspecto, el calculador de segundo desfase está configurado para calcular el

15 segundo desfase para que sea igual a $\left(\left(\left(p_2 + p_0 + 1\right) \gg 1\right) - p_1 + \Delta\right) \gg 1$, donde \gg denota una operación de desplazamiento a la derecha, y el calculador de tercer desfase está configurado para calcular el tercer desfase para que sea igual a $\left(\left(\left(q_2 + q_0 + 1\right) \gg 1\right) - q_1 - \Delta\right) \gg 1$.

En una realización opcional de este aspecto, la unidad de filtro de desbloqueo comprende una unidad de tercer acortamiento configurada para acortar el segundo desfase dentro de un intervalo $-t_{c2}$ y t_{c2} , y acortar el tercer desfase dentro de un intervalo de $-t_{c2}$ y t_{c2} , donde t_{c2} es un valor de umbral que depende del valor de un parámetro de cuantificación asignado al bloque. Este acortamiento opcional por parte de la unidad de tercer acortamiento se alcanza ajustando el segundo o el tercer desfase para que sea igual a $-t_{c2}$ si el segundo o el tercer desfase son menores de $-t_{c2}$ y ajustando el segundo o el tercer desfase para que sea igual a t_{c2} si el segundo o el tercer desfase son mayores de t_{c2} .

En una realización opcional de este aspecto, la unidad de filtro de desbloqueo comprende una unidad de segundo acortamiento configurada para acortar el valor de píxel modificado del píxel siguiente más cercano a la frontera de bloque en la línea de píxeles dentro de un intervalo de cero y un valor máximo definido, y acortar el valor de píxel modificado del píxel siguiente más cercano a la frontera de bloque en la línea de píxeles correspondiente dentro de un intervalo de cero y el valor máximo definido. Este acortamiento opcional por parte de la unidad de segundo acortamiento se consigue ajustando el valor de píxel modificado para que sea igual a cero si el valor de píxel modificado es menor que cero y ajustando el valor de píxel modificado para que sea igual al valor máximo definido si el valor de píxel modificado es mayor que el valor máximo definido.

En una realización opcional de este aspecto, la unidad de filtro de desbloqueo comprende un calculador de cuarto

35 desfase configurado para calcular un cuarto desfase que esté basado en $(p_3 + p_1 - 2 \times p_2 + 2 \times \Delta_{p1}) / 4$, donde p_3 denota un valor de píxel de un píxel tercero siguiente más cercano a la frontera de bloque en la línea de píxeles y Δ_{p1} denota el segundo desfase. La unidad de filtro de desbloqueo comprende también, en esta realización opcional, un calculador de quinto desfase configurado para calcular un quinto desfase para que esté basado en

40 $(q_3 + q_1 - 2 \times q_2 + 2 \times \Delta_{q1}) / 4$, donde q_3 denota un valor de píxel de un píxel tercero siguiente más cercano a la frontera de bloque en la línea de píxeles correspondiente y Δ_{q1} denota el tercer desfase. En esta realización opcional el valor de píxel modificado está configurado para modificar el valor de píxel del píxel segundo siguiente más cercano a la frontera de bloque en la línea de píxeles sumando el cuarto desfase al valor de píxel del píxel segundo siguiente más cercano a la frontera de bloque en la línea de píxeles para formar un valor de píxel modificado, y para modificar el valor de píxel del píxel segundo siguiente más cercano a la frontera de bloque en la línea de píxeles correspondiente sumando el quinto desfase al valor de píxel del píxel segundo siguiente más cercano a la frontera de bloque en la línea de píxeles correspondiente para formar un valor de píxel modificado.

En una realización opcional de este aspecto, el calculador de cuarto desfase está configurado para calcular el cuarto

desfase para que sea igual a $\left(\left(\left(p_3 + p_1 + 1\right) \gg 1\right) - p_2 + \Delta_{p1}\right) \gg 1$, donde \gg denota una operación de desplazamiento a la derecha, y el calculador de quinto desfase está configurado para calcular el quinto desfase para que sea igual a $\left(\left(\left(q_3 + q_1 + 1\right) \gg 1\right) - q_2 + \Delta_{q1}\right) \gg 1$.

Otros aspectos de las realizaciones definen un codificador y un decodificador, respectivamente, que comprenden una unidad de filtro de desbloqueo tal como se ha definido anteriormente.

Además, un aspecto de las realizaciones define un equipo de usuario que comprende una memoria configurada para fotogramas de video codificados almacenados, un descodificador tal como se ha definido anteriormente configurado para descodificar los fotogramas de video en fotogramas de video descodificados y un reproductor de medios configurado para traducir los fotogramas de video descodificados a datos de video que sean reproducibles en una pantalla.

Otro aspecto define un dispositivo de red que se encuentra o que pertenece a un nodo de red en una red de comunicación entre una unidad de transmisión y un equipo de usuario de recepción. El dispositivo de red comprende un codificador y/o un descodificador tal como se ha definido anteriormente.

Otro aspecto de las realizaciones define un programa informático para reducir las aberraciones de bloqueo en una frontera de bloque entre un bloque de píxeles y un bloque de píxeles vecino en un fotograma de video. El programa informático comprende medios de código que, cuando son ejecutados en un ordenador hacen que el ordenador calcule un desfase para que sea igual a $(9 \times (q_0 - p_0) - 3 \times (q_1 - p_1) + 8) \gg 4$, donde \gg denota una operación de desplazamiento a la derecha, p_0 denota un valor de píxel de un píxel más cercano a la frontera de bloque en una línea de píxeles en el bloque, p_1 denota un valor de píxel de un píxel siguiente más cercano a la frontera de bloque en la línea de píxeles, q_0 denota un valor de píxel de un píxel más cercano a la frontera de bloque en una línea de píxeles correspondiente en el bloque vecino y q_1 denota un valor de píxel de un píxel siguiente más cercano a la frontera de bloque en la línea de píxeles correspondiente. La línea de píxeles y la línea de píxeles correspondiente son perpendiculares a la frontera de bloque. Se hace también que el ordenador modifique el valor de píxel del píxel más cercano a la frontera de bloque en la línea de píxeles sumando el desfase al valor de píxel de este píxel. El medio de código hace también que el ordenador modifique el valor de píxel más cercano a la frontera de bloque en la línea de píxeles correspondiente restando el desfase del valor de píxel de este píxel. Cuando se ejecuta en el ordenador, el medio de código hace también que el ordenador acorte el citado primer desfase para que esté dentro de un intervalo de $-t_c$ y t_c ajustando el citado primer desfase para que sea igual a $-t_c$ si el citado primer desfase es menor que $-t_c$ y ajustando el citado primer desfase para que sea igual a t_c si el citado primer desfase es mayor que t_c , donde t_c es un valor de umbral que depende del valor de un parámetro de cuantificación asignado al citado bloque.

Las presentes realizaciones permiten una eficiente reducción en las aberraciones de bloqueo y también tienen buenas características de paso bajo.

Breve descripción de los dibujos

La invención, junto con otros objetos y ventaja de la misma, puede comprenderse mejor haciendo referencia a la siguiente descripción tomada junto con los dibujos que se acompañan, en los cuales:

la Fig. 1 es un diagrama de flujo que ilustra un método de reducir las aberraciones de bloqueo de acuerdo con una realización;

las Figs. 2A y 2B ilustran dos realizaciones de bloques vecinos y una frontera de bloque sobre la cual puede aplicarse el filtrado de desbloqueo;

la Fig. 3 es un diagrama de flujo ilustrativo adicional, etapas opcionales del método de la Fig. 1 de acuerdo con una realización;

la Fig. 4 es un diagrama de flujo que ilustra etapas adicionales, opcionales del método de las Figs. 1 y 5 de acuerdo con una realización;

la Fig. 5 es un diagrama de flujo que ilustra etapas adicionales, opcionales del método de la Fig. 1 de acuerdo con otra realización;

la Fig. 6 es un diagrama de flujo que ilustra etapas adicionales, opcionales del método de la Fig. 5 de acuerdo con una realización;

la Fig. 7 es un diagrama de flujo que ilustra etapas adicionales, opcionales del método de la Fig. 5 de acuerdo con otra realización;

la Fig. 8 es un diagrama de bloques esquemático de una unidad de filtro de desbloqueo de acuerdo con una realización;

la Fig. 9 es un diagrama de bloques esquemático de una unidad de filtro de desbloqueo de acuerdo con otra realización;

la Fig. 10 es un diagrama de bloques esquemático de una implementación de software de un módulo de filtro de desbloqueo en un ordenador de acuerdo con una realización;

la Fig. 11 es un diagrama de bloques esquemático de un codificador de acuerdo con una realización;

la Fig. 12 es un diagrama de bloques esquemático de un descodificador de acuerdo con una realización;

la Fig. 13 es un diagrama de bloques esquemático de un equipo de usuario de acuerdo con una realización; y

la Fig. 14 es una vista general esquemática de una porción de una red de comunicación que comprende un dispositivo de red de acuerdo con una realización.

5 Descripción detallada

En los dibujos, los mismos números de referencia se utilizan para elementos similares o correspondientes.

Las realizaciones se refieren en general al filtrado de desbloqueo para combatir las aberraciones de bloqueo sobre fronteras de bloque en un fotograma de video. Las realizaciones utilizan un filtro de desbloqueo que tiene buenas características de frecuencia. El presente filtro de desbloqueo está configurado para no modificar los valores de píxel sobre una frontera de bloque si estos valores de píxel forman una rampa, sino para ajustar los valores de píxel a una función si por el contrario están en forma de escalones. Esto implica que el filtro de desbloqueo tendrá buenas características de paso bajo y puede eliminar las frecuencias altas que pueden aparecer cerca de la frontera de bloque.

La Fig. 1 es un diagrama de flujo de un método de reducir las aberraciones de bloqueo en una frontera de bloque entre un bloque de múltiples píxeles y un bloque vecino de múltiples píxeles en un fotograma de video de acuerdo con una realización. Como es bien conocido en el sector, un fotograma de video está dividido en bloques de píxeles no superpuestos que son codificados y descodificados de acuerdo con los diferentes modos de intra e inter codificación disponibles. En general, un fotograma de video está dividido en macrobloques no superpuestos de 16 x 16 píxeles. Tal macrobloque puede a su vez estar dividido en bloques más pequeños, tal como de 4 x 4 o de 8 x 8 píxeles. No obstante, serían también posibles bloques rectangulares de acuerdo con las realizaciones, tal como 4 x 8, 8 x 4, 8 x 16 o 16 x 8. Las realizaciones pueden aplicarse a cualquiera de tales bloques de píxeles, incluyendo macrobloques o incluso bloques de píxeles mayores.

En el estándar Codificación de video de alta eficiencia (HEVC – High Efficiency Video Coding, en inglés) emergente, se utilizan las unidades de codificación (CU – Coding Units, en inglés), las unidades de predicción (PU- Prediction Units, en inglés) y las unidades de transformación (TU – Transform units, en inglés). Las unidades de predicción están definidas dentro de una unidad de codificación y contienen los modos de intra o inter predicción. Las unidades de transformación están definidas en una unidad de codificación y el mayor tamaño de transformación es 32 x 32 píxeles y el menor tamaño es 4 x 4 píxeles. El tamaño de CU varía actualmente de 64 x 64 píxeles (el más grande) a 4 x 4 píxeles (el más pequeño). De esta manera, la CU más grande puede dividirse en CU más pequeñas, dependiendo el “nivel de granularidad” de las características locales del fotograma. Esto significa que la CU más grande puede dividirse en CU más pequeñas de diferentes tamaños. Las realizaciones pueden utilizarse también junto con tales unidades de codificación, que se consideran contenidas en la expresión “bloque de píxeles” tal como se utiliza en esta memoria.

Cada píxel del bloque tiene un valor de píxel respectivo. Los fotogramas de video generalmente tienen valores de color asignados a los píxeles, donde los valores de color se representan en unos formatos de color definidos. Uno de los formatos de color comunes utiliza una componente de luminancia y dos componentes de crominancia para cada píxel, aunque existen otros formatos, tales como los que utilizan componentes rojos, verdes y azules para cada píxel.

Tradicionalmente, el filtrado de la componente de luminancia y de la componente de crominancia se realiza separadamente, posiblemente empleando diferentes decisiones de filtrado y diferentes filtros de desbloqueo. Es, por lo tanto, posible, que las decisiones de filtrado de luminancia se utilicen en filtrado croma, como en HEVC. Las realizaciones pueden aplicarse para filtrar la componente de luminancia, filtrar la componente de crominancia o filtrar tanto la componente de luminancia como la componente de crominancia. En una realización particular, las realizaciones se aplican para conseguir el filtrado de luminancia o de luma. Decisiones de filtrado, o partes de las decisiones de filtrado para una componente, tal como luma, pueden utilizarse entonces cuando se toman las decisiones de filtrado para otras componentes, tales como croma.

El filtrado de desbloqueo se lleva a cabo sobre una frontera, borde o límite entre bloques vecinos. En consecuencia, tales fronteras pueden ser fronteras verticales 1, véase la Fig. 2A, entre dos bloques vecinos 10, 20 presentes a cada lado en el fotograma de video. Si no, las fronteras son fronteras horizontales 1, véase la Fig. 2B, entre dos bloques vecinos 10, 20, donde ningún bloque 10 está situado por encima del otro bloque 20 en el fotograma de video. En una realización particular, las fronteras verticales son filtradas primero empezando desde la frontera más a la izquierda y continuando a través de las fronteras hacia el lado derecho en su orden geométrico. A continuación, las fronteras horizontales son filtradas empezando con la frontera en el lado superior y continuando a través de las fronteras hacia abajo en su orden geométrico. Las realizaciones, no obstante, no están limitadas a este orden de filtrado particular, y pueden de hecho aplicarse a cualquier orden de filtrado predefinido. En una realización particular, las fronteras en el borde del fotograma de video preferiblemente no se filtran y por ello quedan excluidas del filtrado de desbloqueo.

El método de esta realización se inicia en la etapa S1, en la que se calcula el primer desfase o delta para una línea 12 de píxeles 11, 13, 15, 17 en el bloque 10. De acuerdo con las realizaciones, este primer desfase se calcula

basado en $(9 \times (q_0 - p_0) - 3 \times (q_1 - p_1)) / 16$, donde p_0 denota el valor de píxel del píxel 11 más cercano a la frontera de bloque 1 en la línea 12 de píxeles 11, 13, 15, 17 del bloque 10, p_1 denota el valor de píxel del píxel 13 siguiente más cercano a la frontera de bloque 1 en la línea 12 de píxeles 11, 13, 15, 17, q_0 denota el valor de píxel del píxel 21 más cercano a la frontera de bloque 1 en una línea correspondiente u opuesta 22 de píxeles 21, 23, 25, 27 en un bloque vecino 20 y q_1 denota el valor de píxel del píxel 23 siguiente más cercano a la frontera de bloque 1 en la línea 22 correspondiente de píxeles 21, 23, 25, 27.

La línea 12 de píxeles 11, 13, 15, 17 del bloque 10 y la línea 22 correspondiente de píxeles 21, 23, 25, 27 del bloque vecino 20 pertenecen a la misma línea de píxeles horizontal, es decir a la fila de píxeles que se extiende sobre una frontera vertical 1, véase la Fig. 2A, o pertenecen a la misma línea de píxeles vertical, es decir, a la columna de píxeles que se extiende sobre una frontera horizontal 1, véase la Fig. 2B. Por ello, la línea 12 de píxeles 11, 13, 15, 17 y la línea 22 correspondiente de píxeles 21, 23, 25, 27 son perpendiculares a la frontera de bloque 1 entre el bloque 10 y el bloque vecino 20. Además, la línea 12 de píxeles 11, 13, 15, 17 en el bloque 10 y la línea 22 correspondiente de píxeles 21, 23, 25, 27 en el bloque vecino 20 tienen el mismo número de línea. Por ejemplo, si el bloque 10 y el bloque vecino 20 comprenden cada uno N , tal como ocho, filas o columnas de píxeles, siendo los números de fila o columna $i = 0 \dots N - 1$ entonces la línea 10 de píxeles 11, 13, 15, 17 tiene el número de línea i , pero en el bloque vecino 20. Así, la línea 12 de píxeles 11, 13, 15, 17 en el bloque y la línea 22 correspondiente de píxeles 21, 23, 25, 27 en el bloque vecino 20 son líneas opuestas con respecto a la frontera de bloque 1.

De acuerdo con las realizaciones, los términos “línea de píxeles” y línea de píxeles correspondiente” se emplean para denotar una “fila de píxeles” y una “fila de píxeles correspondiente” en el caso de una frontera de bloque vertical como en la Fig. 2A, y denota una “columna de píxeles” y una “columna de píxeles correspondiente” en el caso de una frontera de bloque horizontal como en la Fig. 2B.

Una siguiente etapa S2 modifica el valor de píxel del píxel 11 más cercano a la frontera de bloque 1 en la línea 12 de píxeles 11, 13, 15, 17 sumando el primer desfase calculado en la etapa S1 al valor de píxel de este píxel 11 para formar un valor de píxel modificado p_0' . Así, el valor de píxel modificado p_0' para este píxel 11 es $p_0' = p_0 + \Delta$, donde Δ denota el primer desfase de la etapa S1. Correspondientemente, el valor de píxel del píxel 21 más cercano a la frontera de bloque 1 pero en la línea 22 correspondiente de píxeles 21, 23, 25, 27 en el bloque vecino 20 se modifica en la etapa S3 restando el primer desfase del valor de píxel 21 para formar un valor de píxel modificado q_0' . El valor de píxel modificado q_0' se calcula por lo tanto como $q_0' = q_0 - \Delta$.

La modificación del valor de píxel 11 más cercano a la frontera de bloque 1 en la línea 12 de píxeles 11, 13, 15, 17 en el bloque 10, en la etapa S2 y la modificación del valor de píxel para el píxel 21 más cercano a la frontera de bloque 1 en la línea 22 correspondiente de píxeles 21, 23, 25, 27 en el bloque vecino 20, en la etapa S3 puede ser llevada a cabo en serie, como se ilustra en la Fig. 1, o en serie pero en orden inverso, es decir, la etapa S3 antes que la etapa S2. Si no, las dos etapas S2 y S3 pueden ser ejecutadas al menos parcialmente en paralelo.

El método ilustrado en la Fig. 1 y que incluye las etapas S1 a S3 reduce las aberraciones de bloqueo en la frontera de bloque 1 procesando los píxeles en una línea 11 de píxeles 11, 13, 15, 17 en el bloque y también en la línea 22 correspondiente de píxeles 21, 23, 25, 27 en el bloque vecino 20. Esta modificación de píxel puede ser llevada a cabo para una de las líneas (horizontales o verticales) 12 del bloque 10 o para varias, es decir, por lo menos dos, líneas 12 en el bloque 10, posiblemente para todas las líneas 12 (horizontales o verticales) en el bloque 10. Esto se ilustra esquemáticamente mediante la línea L1.

En general, y como se ha explicado en lo anterior, el filtrado de desbloqueo implica decisiones de filtro que determinan si debe aplicarse un filtro de desbloqueo sobre una frontera de bloque horizontal o vertical para un bloque. Si tal decisión es positiva, el filtro de desbloqueo puede entonces aplicarse a todas las columnas (líneas verticales) o filas (líneas horizontales) del bloque. Si no, se toman otras decisiones sobre el filtrado para cada columna o fila para decidir si debe aplicarse o no el filtro de desbloqueo y/o qué tipo de filtro de desbloqueo utilizar para esa columna o fila particulares. Por lo tanto, el método que se ilustra en la Fig. 1 podría aplicarse a una columna, a una fila, a múltiples columnas, a múltiples filas o a ambas para al menos una columna y al menos una fila en un bloque de píxeles en el fotograma de video. Se prevé además que el filtrado de desbloqueo no necesariamente tenga que ser aplicado a todos los bloques del fotograma de video. En claro contraste, tal filtrado de desbloqueo se aplica preferiblemente solo a los bloques y sobre fronteras de bloques en las que existen aberraciones de bloqueo según se determina mediante una o más de las decisiones de filtrado.

El filtrado de desbloqueo se describirá ahora con más detalle en esta memoria en relación con varias realizaciones de implementación.

El acortamiento del primer desfase puede realizarse en una realización para restringir con ello el valor del primer desfase para que esté dentro del intervalo de $-t_c$ y t_c . El valor de umbral t_c depende preferiblemente de un valor del parámetro de cuantificación (QP) asignado al bloque. En tal caso, podría utilizarse una tabla de diferentes valores de umbral para diferentes valores QP. La Tabla 1 que se incluye a continuación es un ejemplo de tal tabla.

Tabla 1 – Valores de t_c para diferentes valores de QP

QP	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
t_c	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
QP	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37
t_c	1	1	1	1	1	1	1	1	2	2	2	2	3	3	3	3	4	4	4
QP	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	
t_c	5	5	6	6	7	8	9	9	10	10	11	11	12	12	13	13	14	14	

5 Las presentes realizaciones no están, no obstante, limitadas a la relación particular entre t_c y QP según se muestra en la Tabla 1, sino que podrían por el contrario determinar de algún otro modo el valor de t_c para el bloque sobre la base del valor de QP para el bloque.

10 La Fig. 3 ilustra esta acción de acortamiento. El método continúa entonces desde la etapa S1 de la Fig. 1. Una etapa S10 siguiente analiza si el primer desfase está dentro del intervalo $-t_c$ y t_c , es decir, si $-t_c \leq \Delta \leq t_c$. En tal caso, el método continúa hacia la etapa S2 de la Fig. 1 y no se requiere ninguna otra modificación del primer desfase. No obstante, si el primer desfase no está dentro del intervalo según se determina en la etapa S10, el método continúa hacia la etapa S11, donde el primer desfase es acortado para que tenga un valor dentro del intervalo. Así, si $\Delta < -t_c$, el primer desfase se ajusta para tener un valor de $-t_c$ en la etapa S11. Correspondientemente, si $\Delta > t_c$ entonces el primer desfase se ajusta para que tenga un valor de t_c en la etapa S11. El método continúa entonces hacia la etapa S2 de la Fig. 2.

15 Correspondientemente, los valores de píxel modificados calculados en las etapas S2 y S3 de la Fig. 1 podrían ser acortados para que estén dentro de un intervalo permitido. Esto se ilustra esquemáticamente mediante el diagrama de flujo de la Fig. 4. El método continúa entonces desde la etapa S2 o S3 de la Fig. 1, y una etapa siguiente S20 analiza si el valor de píxel modificado, es decir, p_0' o q_0' está dentro de un intervalo permitido de 0 y M. El parámetro M denota un valor máximo definido que el valor de píxel modificado puede asumir. En una realización particular, este valor máximo se define basándose en el número de bits que se ocupan en los valores de píxel. Así, asúmase que
 20 los valores de píxel están en la forma de un valor de m bits; entonces M es preferiblemente igual a $2^m - 1$. Por ejemplo, si cada valor de píxel es un valor de 8 bits, es decir, $m = 8$, entonces $M = 255$. La etapa S20 analiza por lo tanto si $0 \leq p_0' \leq M$ o si $0 \leq q_0' \leq M$. Si el valor de píxel modificado está dentro del intervalo $[0, M]$ el método continúa hacia la etapa S3 de la Fig. 1 o finaliza. No obstante, si el valor de píxel modificado está fuera del intervalo, se acorta en la etapa S21 para que tenga un valor dentro del intervalo. En otras palabras, si $p_0' < 0$ o $q_0' < 0$, entonces el valor de píxel modificado se ajusta en la etapa S21 a $p_0' = 0$ o $q_0' = M$. Correspondientemente, si $p_0' > M$ o $q_0' > M$, entonces la etapa S21 ajusta el valor de píxel modificado a $p_0' = M$ o $q_0' = M$.

30 La etapa S1 de la Fig. 1 calcula el primer desfase para que se base en $(9 \times (q_0 - p_0) - 3 \times (q_1 - p_1)) / 16$. En una realización particular, el primer desfase se calcula en la etapa S1 para que sea igual a $(9 \times (q_0 - p_0) - 3 \times (q_1 - p_1)) / 16$. En una realización, los valores de píxel se actualizan por lo tanto mediante el filtro de desbloqueo propuesto utilizando los siguientes cálculos:

$$\begin{aligned} \Delta &= (9 \times (q_0 - p_0) - 3 \times (q_1 - p_1)) / 16 \\ p_0' &= p_0 + \Delta \\ q_0' &= q_0 - \Delta \end{aligned}$$

Puede utilizarse también el acortamiento del primer desfase y/o de los valores de píxel modificados como se ilustra en las Figs. 3 y 4.

35 En una realización alternativa, el desfase se calcula en función de $(9 \times (q_0 - p_0) - 3 \times (q_1 - p_1)) / 16$. Tal función podría entonces definirse de manera que el cálculo del primer desfase se lleve a cabo de manera eficiente mediante hardware. En tal caso, se prefiere generalmente no tener divisiones y/o definir la función de manera que el primer desfase sea un valor entero. En una realización, $(X + 8) \gg 4$ se utiliza como una expresión entera de $X / 16$, donde \gg denota una operación de desplazamiento a la derecha. Así, en una realización particular la etapa S1 calcula el primer desfase para estar basado en y que sea preferiblemente igual a $(9 \times (q_0 - p_0) - 3 \times (q_1 - p_1) + 8) \gg 4$.

En esta realización, los valores de píxel se actualizan por lo tanto mediante el desbloqueo de filtro propuesto utilizando los siguientes cálculos:

$$\begin{aligned}\Delta &= (9 \times (q_0 - p_0) - 3 \times (q_1 - p_1) + 8) \gg 4 \\ p_0' &= p_0 + \Delta \\ q_0' &= q_0 - \Delta\end{aligned}$$

o si se utiliza acortamiento:

$$\begin{aligned}\Delta &= \text{Clip3}(-t_C, t_C, (9 \times (q_0 - p_0) - 3 \times (q_1 - p_1) + 8) \gg 4) \\ p_0' &= \text{Clip}(p_0 + \Delta) \\ q_0' &= \text{Clip}(q_0 - \Delta)\end{aligned}$$

5 donde Clip3 (A, B, x) se define como Clip3 (A, B, x) = A si $x < A$, Clip3 (A, B, x) = B si $x > B$ y Clip3 (A, B, x) = x si $A \leq x \leq B$ y Clip (x) se define como Clip (0, M, x).

En realizaciones alternativas se utilizan otras implementaciones, tal como representaciones enteras de $(9 \times (q_0 - p_0) - 3 \times (q_1 - p_1)) / 16$ y preferiblemente representaciones enteras tales que se implementan eficientemente en hardware.

Las realizaciones descritas anteriormente definen un filtro de desbloqueo que genera un primer desfase para los píxeles más cercanos a la frontera de bloque con una fórmula que da como resultado aproximadamente cero, preferiblemente exactamente cero, cuando se aplica a una rampa de valores de píxel y que da como resultado un valor de desfase que aproxima un escalón en los valores de píxel cuando se aplica a un escalón de valores de píxel. Por ejemplo, una rampa puede describirse como valores de píxel linealmente crecientes o decrecientes, por ejemplo, 10, 20, 30, 40. Cuando se calcula el primer desfase para estos valores de píxel, es decir, $p_1 = 10$, $p_0 = 20$, $q_0 = 30$, $q_1 = 40$, el primer desfase será cero. Correspondientemente, un escalón puede describirse como un aumento o disminución en escalón en los valores de píxel, por ejemplo, 10, 10, 20, 20. Cuando se calcula el primer desfase para estos valores de píxel, es decir, $p_1 = 10$, $p_0 = 10$, $q_0 = 20$, $q_1 = 20$, el primer desfase será 3,75 si $\Delta = (9 \times (q_0 - p_0) - 3 \times (q_1 - p_1)) / 16$ o 4 si $\Delta = (9 \times (q_0 - p_0) - 3 \times (q_1 - p_1) + 8) \gg 4$. Los valores de píxel modificados serán entonces 10, 13,75, 16,25, 30 o 10, 14, 16, 20 y por ello se consigue una aproximación del escalón. El primer desfase es también cero para una línea plana, es decir, si los valores de píxel son iguales a $p_0 = p_1 = q_0 = q_1$.

En una realización particular, también los valores de píxel de los píxeles siguientes más cercanos a la frontera de bloque pueden ser modificados. Esto se describe con más detalle en esta memoria con referencia a las Figs. 2A, 2B y 5. El método continúa entonces desde la etapa S3 de la Fig. 1 o desde la etapa S2, o realmente desde la etapa S1. Una siguiente etapa S30 calcula un segundo desfase para que esté basado en $(p_2 + p_0 - 2 \times p_1 + 2 \times \Delta) / 4$, donde p_2 denota el valor de píxel del píxel 15 segundo siguiente más cercano a la frontera de bloque 1 en la línea 12 de píxeles 11, 13, 15, 17 del bloque 10. Una siguiente etapa S31 calcula un tercer desfase para que se base en $(q_2 + q_0 - 2 \times q_1 - 2 \times \Delta) / 4$, donde q_2 denota el valor de píxel del píxel 25 segundo siguiente más cercano a la frontera de bloque 1 en la línea 22 correspondiente de píxeles 21, 23, 25, 27. Las etapas S30 y S31 pueden ser realizadas secuencialmente en cualquier orden o al menos parcialmente en paralelo.

El segundo desfase calculado en la etapa S30 se utiliza en la etapa S32 para modificar el valor de píxel del píxel 13 siguiente más cercano a la frontera de bloque 1 en la línea 12 de píxeles 11, 13, 15, 17. En una realización, el segundo desfase se suma al valor de píxel de este píxel 13 para obtener un valor de píxel modificado. Correspondientemente, la etapa S33 modifica el valor de píxel del píxel 23 siguiente más cercano a la frontera de bloque 1 en la línea 22 correspondiente de píxeles 21, 23, 25, 27 sumando el tercer desfase al valor de píxel de este píxel 23 para formar un valor de píxel modificado. Las etapas S32 y S33 pueden ser ejecutadas en cualquier orden secuencial o al menos parcialmente en paralelo.

El filtro de desbloqueo propuesto es, en esta realización, un filtro de paso bajo en todas las posiciones de filtro p_0 , p_1 , q_0 , q_1 . Cuando se aplica a una señal de rampa, el filtro de desbloqueo no la modificará puesto que Δ , Δp , Δq serán todos igual a cero, donde Δp denota el segundo desfase y Δq denota el tercer desfase. Cuando se aplica a una señal de escalón, es decir, a una aberración de desbloqueo, el filtro de desbloqueo la aproximará, es decir, reducirá la aberración de bloque. En claro contraste con el filtro HEVC actual, el filtro de desbloqueo propuesto aproximará y por ello atenuará pequeñas ondulaciones a los lados de la frontera de bloque.

En la realización descrita anteriormente, los filtros para los coeficientes que se encuentran en la segunda posición con respecto a la frontera de bloque (p_1 , q_1) se producen utilizando una combinación del primer desfase y un filtro

de paso bajo simétrico. Una ventaja de esta realización es que los desfases para los píxeles segundos siguientes más cercanos a la frontera de bloque pueden tener diferentes valores que permiten una mejor adaptación local.

De manera similar al primer desfase, los desfases segundo y tercero pueden ser acertados para que estén dentro de un intervalo de $-t_{C2}$ y t_{C2} , donde el valor de umbral t_{C2} se determina sobre la base del valor de QP asignado al bloque. En una realización particular el valor de umbral t_{C2} se determina sobre la base del valor de umbral t_C utilizado

para el acortamiento del primer desfase en la Fig. 3. Por ejemplo, $t_{C2} = \frac{t_C}{2}$ o con una implementación adaptada de hardware $t_{C2} = t_C \gg 1$.

La Fig. 6 es un diagrama de flujo que ilustra tal acortamiento de los desfases segundo y tercero. El método continúa desde la etapa S30 o S31 de la Fig. 5. Una etapa S40 siguiente analiza si el desfase segundo o tercero está dentro del intervalo de $-t_{C2}$ y t_{C2} . Si esto es verdadero el método continúa hacia la etapa S32 o S33 de la Fig. 5. No obstante, si el desfase segundo o tercero es menor que $-t_{C2}$ o mayor que t_{C2} el método continúa hacia la etapa S41. Esta etapa S41 acorta el desfase segundo o tercero para que sea $-t_{C2}$ si $\Delta_{p,q} < -t_{C2}$ o ajusta el desfase segundo o tercero a t_{C2} si $\Delta_{p,q} > t_{C2}$.

Los valores de píxel modificados siguientes más cercano a la frontera de bloque son preferiblemente acertados para que estén en el intervalo de 0 a M de manera similar a los valores de píxel modificados para los píxeles más cercanos a la frontera de bloque. Por ello, las etapas del método de la Fig. 4 pueden aplicarse también a estos píxeles para forzar a los valores de píxel a estar dentro de $[0, M]$.

En una realización, los valores de píxel modificados de los píxeles 13, 23 siguientes más cercanos a la frontera de bloque 1 se calculan como:

$$\begin{aligned}\Delta_p &= (p2 + p0 - 2 \times p1 + 2 \times \Delta) / 4 \\ p1' &= p1 + \Delta_p \\ \Delta_q &= (q2 + q0 - 2 \times q1 - 2 \times \Delta) / 4 \\ q1' &= q1 + \Delta_q\end{aligned}$$

Opcionalmente se utiliza el acortamiento de los desfases segundo y tercero y/o los valores de píxel modificados tal como se ha descrito anteriormente.

En una realización alternativa, los cálculos de los desfases podrían ser realizados para ser independientes uno de otro.

$$\begin{aligned}\Delta &= (9 \times (q0 - p0) - 3 \times (q1 - p1) + 8) \gg 4 \\ p0' &= p0 + \Delta \\ q0' &= q0 - \Delta \\ \Delta_p &= (p0 + p2 - 2 \times p1) / 4 \\ p1' &= p1 + \Delta_p + \Delta / 2 \\ \Delta_q &= (q0 + q2 - 2 \times q1) / 4 \\ q1' &= q1 + \Delta_q - \Delta / 2\end{aligned}$$

Matemáticamente esto es equivalente a la realización descrita previamente. Puede también realizarse un acortamiento opcional para esta realización.

En una realización alternativa, los desfases segundo y tercero se calculan sobre la base, tal como una función, de $(p2 + p0 - 2 \times p1 + 2 \times \Delta) / 4$ y $(q2 + q0 - 2 \times q1 - 2 \times \Delta) / 4$, respectivamente. Por ejemplo, tal función podría ser adaptada para una implementación de hardware, y/o ser una representación entera de $(p2 + p0 - 2 \times p1 + 2 \times \Delta) / 4$ y $(q2 + q0 - 2 \times q1 - 2 \times \Delta) / 4$. Un ejemplo particular de tal implementación de hardware que produce valores enteros es calcular el segundo desfase para que esté basado en, sea preferiblemente igual a, $((((p2 + p0 + 1) \gg 1) - p1 + \Delta) \gg 1)$. Correspondientemente, el tercer desfase se calcula preferiblemente para que esté basado en, sea preferiblemente igual a $((((q2 + q0 + 1) \gg 1) - q1 - \Delta) \gg 1)$.

En tal caso, los valores de píxel modificados se calculan como:

$$\Delta_p = (((p2 + p0 + 1) \gg 1) - p1 + \Delta) \gg 1$$

$$p1' = p1 + \Delta_p$$

$$\Delta_q = (((q2 + q0 + 1) \gg 1) - q1 - \Delta) \gg 1$$

$$q1' = q1 + \Delta_q$$

o si se utiliza el acortamiento:

$$\Delta_p = Clip3(-t_{C2}, t_{C2}, (((p2 + p0 + 1) \gg 1) - p1 + \Delta) \gg 1)$$

$$p1' = Clip(p1 + \Delta_p)$$

$$\Delta_q = Clip3(-t_{C2}, t_{C2}, (((q2 + q0 + 1) \gg 1) - q1 - \Delta) \gg 1)$$

$$q1' = Clip(q1 + \Delta_q)$$

5 Una realización alternativa para el cálculo de los desfases segundo y tercero, que está adaptada para implementación mediante hardware, es:

$$\Delta_p = Clip3(-t_{C2}, t_{C2}, ((p2 + p0 - ((p1 - \Delta) \ll 1) + 2) \gg 2))$$

$$\Delta_q = Clip3(-t_{C2}, t_{C2}, ((q2 + q0 - ((q1 + \Delta) \ll 1) + 2) \gg 2))$$

donde \ll denota una operación de desplazamiento a la izquierda.

Otra realización alternativa para el cálculo de los desfases segundo y tercero, que está adaptada para implementación mediante hardware, es:

$$10 \quad \Delta_p = Clip3(-t_{C2}, t_{C2}, (((p2 + p0 + 1) \gg 1) - p1 + \Delta + 1) \gg 1))$$

$$\Delta_q = Clip3(-t_{C2}, t_{C2}, (((q2 + q0 + 1) \gg 1) - q1 - \Delta + 1) \gg 1))$$

En otra realización, puede utilizarse un filtro de paso bajo más fuerte en combinación con el primer desfase. En este

caso el segundo desfase se calcula sobre la base de, tal como igual a $(p2 + p0 - 2 \times p1 + \Delta) / 2$ o, en una implementación adaptada mediante hardware, sobre la base de, preferiblemente igual a $((p2 + p0 + 1 + \Delta) \gg 1) - p1$ o si no a $((p2 + p0 + \Delta - (p1 \ll 1) + 1) \gg 1)$. El tercer desfase podría entonces

15 calcularse sobre la base de, tal como igual a $(q2 + q0 - 2 \times q1 - \Delta) / 2$ o, en una implementación adaptada a hardware, sobre la base de, preferiblemente igual a $((q2 + q0 + 1 - \Delta) \gg 1) - q1$ o si no, $((q2 + q0 - \Delta - (q1 \ll 1) + 1) \gg 1)$.

20 Una realización particular calcula los desfases primero, segundo y tercero para una línea de píxeles en el bloque para modificar con ello los valores de píxel tanto de los píxeles más cercanos a la frontera de bloque como de los píxeles siguientes más cercanos a la frontera de bloque.

En una realización alternativa, se toma primero una decisión acerca de si calcular los desfases segundo y tercero y con ello si modificar los valores de píxel de los píxeles siguiente más cercanos a la frontera de bloque además de modificar los valores de píxel de los píxeles más cercanos a la frontera de bloque.

25 De manera similar a las realizaciones descritas en el anterior cálculo de los desfases segundo y tercero para modificar los valores de píxel de los píxeles siguientes más cercanos a la frontera de bloque pueden calcularse desfases cuarto y quinto para modificar los valores de píxel de los píxeles siguientes más cercanos a la frontera de bloque.

La Fig. 7 es un diagrama de flujo que ilustra tal realización. El método continúa desde la etapa S33 de la Fig. 5. Una

30 siguiente etapa S50 calcula un cuarto desfase sobre la base de $(p3 + p1 - 2 \times p2 + 2 \times \Delta_{p1}) / 4$, donde p3 denota el valor de píxel del píxel 17 tercero siguiente más cercano a la frontera de bloque 1 en la línea 12 de píxeles 11, 13, 15, 17 en el bloque 10 y Δ_{p1} denota el segundo desfase. La etapa S51 correspondientemente calcula un quinto

desfase para que esté basado en $(q3 + q1 - 2 \times q2 + 2 \times \Delta_{q1}) / 4$, donde q3 denota el valor de píxel del píxel 27 tercero más cercano a la frontera de bloque 1 en la línea 22 correspondiente de píxeles 21, 23, 25, 27 en el bloque vecino 20 y Δ_{q1} denota el tercer desfase. Las etapas S50 y S51 pueden ser ejecutadas secuencialmente en cualquier

35 orden o al menos parcialmente en paralelo.

Los desfases cuarto y quinto pueden, en una realización alternativa, calcularse para estar basados en $(p3+p1-2 \times p2+\Delta_{p1})/2$ y $(q3+q1-2 \times q2+\Delta_{q1})/2$ respectivamente, o si no, en $(p3+p1-2 \times p2+2 \times \Delta_{q1})/4$ y $(q3+q1-2 \times q2+2 \times \Delta_{p2})/4$.

5 Las siguientes dos etapas modifican los valores de píxel sobre la base de los desfases cuarto y quinto. Así, la etapa S52 modifica el valor de píxel del píxel 15 segundo siguiente más cercano a la frontera de bloque 1 en la línea 12 de píxeles 11, 13, 15, 17 del bloque 10 sumando el cuarto desfase al valor de píxel de este píxel 15 para formar un valor de píxel modificado. La etapa S53 modifica de manera correspondiente el valor de píxel del píxel 25 segundo siguiente más cercano a la frontera de bloque 1 en la línea 22 correspondiente de píxeles 21, 23, 25, 27 en el bloque vecino 20 sumando el quinto desfase al valor de píxel de este píxel 25 para formar un valor de píxel modificado. Las etapas S52 y S53 pueden ser ejecutadas de manera secuencial en cualquier orden o al menos parcialmente en paralelo.

10 Este concepto puede generalizarse también para los filtros que modifican más de tres píxeles desde la frontera de bloque. Por ejemplo, el cuarto píxel 17, 27 desde la frontera de bloque 1 puede obtenerse utilizando una combinación de los desfases para los píxeles terceros 15, 25 (o el desfase de los píxeles segundos 13, 23 o el desfase de los píxeles primeros 11, 21) y un filtro de paso bajo simétrico. También son posibles filtros incluso mayores.

15 De manera similar a las realizaciones previas, los desfases cuarto y quinto pueden ser también acortados. En tal caso, podría utilizarse el mismo intervalo que para los desfases segundo y tercero o la mitad de ese intervalo. En una realización alternativa, el intervalo de acortamiento es $-t_{c3}$ a t_{c3} , donde el valor de umbral t_{c3} se determina sobre la base del valor de QP asociado con el bloque. También pueden acortarse los valores de píxel modificados de los píxeles segundos siguientes más cercanos a la frontera de bloque para que estén dentro del intervalo de 0 a M, como se describe en la Fig. 4.

En una realización particular, la modificación de estos valores de píxel se lleva a cabo de acuerdo con lo que sigue:

$$\begin{aligned} \Delta_{p2} &= (p3+p1-2 \times p2+2 \times \Delta_{p1})/4 \\ p2' &= p2+\Delta_{p2} \\ \Delta_{q2} &= (q3+q1-2 \times q2+2 \times \Delta_{q1})/4 \\ q2' &= q2+\Delta_{q2} \end{aligned}$$

25 donde Δ_{p1} , Δ_{q2} denotan los desfases cuarto y quinto, respectivamente. Opcionalmente puede realizarse un acortamiento tal como se describe en lo que sigue.

Una representación de $(p3+p1-2 \times p2+2 \times \Delta_{p1})/4$ y $(q3+q1-2 \times q2+2 \times \Delta_{q1})/4$ adecuada para implementación mediante hardware puede ser utilizada para obtener los desfases cuarto y quinto como valores enteros. En una realización particular, el cuarto desfase se calcula para estar basado en, ser preferiblemente igual a $((p3+p1+1) \gg 1) - p2 + \Delta_{p1} \gg 1$. Correspondientemente, $((q3+q1+1) \gg 1) - q2 + \Delta_{q1} \gg 1$ es una representación entera para el quinto desfase.

35 Las presentes realizaciones están adaptadas para combatir aberraciones de bloqueo en conexión con la codificación y la decodificación de fotogramas de video. Las realizaciones pueden aplicarse por lo tanto a estándares de codificación y decodificación tales que dividen los fotogramas de video en bloques de píxeles y por ello se corre el riesgo de tener aberraciones de bloqueo sobre las fronteras de bloque. Ejemplos de tales estándares, a los cuales pueden aplicarse las presentes realizaciones, son H.264 y HEVC. En particular HEVC tiene la posibilidad de seleccionar entre un modo de filtrado débil y un modo de filtrado fuerte. Las presentes realizaciones pueden entonces ventajosamente utilizarse en el modo de filtrado débil para calcular los desfases que se emplean para modificar los valores de píxel en una línea de píxeles y una línea de píxeles correspondiente sobre una frontera de bloque. Por ello, las decisiones sobre si efectuar tal filtrado de desbloqueo de acuerdo con el HEVC de la técnica anterior pueden ser utilizadas ventajosamente para las presentes realizaciones.

45 Un aspecto particular se refiere a un método de reducir las aberraciones de bloqueo en una frontera de bloque entre un bloque de múltiples píxeles y un bloque vecino de múltiples píxeles en un fotograma de video. Con referencia a la Fig. 1, el método comprende calcular, en la etapa S1, un primer desfase sobre la base de un valor de píxel de un píxel más cercano a la frontera de bloque en una línea de píxeles en el bloque, un valor de píxel de un píxel siguiente más cercano a la frontera de bloque en la línea de píxeles, un valor de píxel de un píxel más cercano a la frontera de bloque en una línea de píxeles correspondiente u opuesta en el bloque vecino y un valor de píxel de un píxel siguiente más cercano a la frontera de bloque en la línea de píxeles correspondiente. La línea de píxeles y la

línea de píxeles vecina son perpendiculares a la frontera de bloque. El primer desfase se calcula sobre la base de estos valores de píxel y una fórmula que produce un valor de desfase que se aproxima a cero, preferiblemente que es igual a cero, cuando los valores de píxel crecen y decrecen linealmente o son iguales cuando se desplazan a lo largo de la línea de píxeles y de la línea de píxeles correspondiente y resulta en un valor de desfase que aproxima un escalón en los valores de píxel cuando los valores de píxel crecen y decrecen en escalones cuando se desplazan a lo largo de la línea de píxeles y de la línea de píxeles correspondiente. El método comprende también modificar, en la etapa S2, el valor de píxel del píxel más cercano a la frontera de bloque en la línea de píxeles sumando el primer desfase al valor de píxel del píxel más cercano a la frontera de bloque en la línea de píxeles. Una etapa siguiente S3 modifica el valor de píxel del píxel más cercano a la frontera de bloque en la línea de píxeles correspondiente restando el primer desfase del valor de píxel del píxel más cercano a la frontera de bloque en la línea de píxeles correspondiente.

Este aspecto particular puede ser combinado con las realizaciones explicadas en lo anterior con respecto a las Figs. 1 - 7.

El método de reducir las aberraciones de bloqueo de acuerdo con la realización descrita en la Fig. 1 se lleva a cabo preferiblemente mediante una unidad de filtro de desbloqueo. Por ello, tal unidad de filtro de desbloqueo calcula entonces el primer desfase en la etapa S1 y modifica los valores de píxel de los píxeles más cercanos a la frontera de bloque en las etapas S2 y S3. La Fig. 8 es un diagrama de bloques esquemático de una realización de tal unidad de filtro de desbloqueo 100.

La unidad de filtro de desbloqueo 100 comprende un calculador de primer desfase 110 configurado para calcular un primer desfase sobre la base de $(9 \times (q_0 - p_0) - 3 \times (q_1 - p_1)) / 16$, como se ha explicado previamente en esta memoria para una línea de píxeles en un bloque de píxeles en un fotograma de video. Un modificador 120 de píxel de la unidad de filtro de desbloqueo 100 está configurado para modificar el valor de píxel de un píxel más cercano a una frontera de bloque en la línea de píxeles en el bloque sumando el primer desfase calculado mediante el calculador de primer desfase 110 al valor de píxel de este píxel para formar un valor de píxel modificado. El modificador 120 de píxel modifica el valor de píxel de un píxel más cercano a la frontera de bloque pero en una línea de píxeles correspondiente en un bloque vecino de píxeles en el fotograma de video. Esta modificación por parte del modificador 120 de valor de píxel se consigue restando el primer desfase calculado por el calculador de primer desfase 110 del valor de píxel de este píxel para formar un valor de píxel modificado.

Por ello, en una realización particular el modificador 120 de píxel está configurado para sumar el primer desfase al valor de píxel del píxel más cercano a la frontera de bloque en la línea de píxeles para formar un valor de píxel modificado. El modificador 120 de píxel está además configurado para restar el primer desfase del valor de píxel del píxel más cercano a la frontera de bloque en la línea de píxeles correspondiente para formar un valor de píxel modificado.

En una realización particular, el calculador 110 de primer desfase está configurado para calcular el primer desfase para que sea $f((9 \times (q_0 - p_0) - 3 \times (q_1 - p_1)) / 16)$, es decir, una función $f(\)$ o representación de $(9 \times (q_0 - p_0) - 3 \times (q_1 - p_1)) / 16$. Esta función preferiblemente proporciona una representación entera de $(9 \times (q_0 - p_0) - 3 \times (q_1 - p_1)) / 16$ y preferiblemente la función es adecuada para implementación mediante hardware. El calculador de primer desfase 110 está configurado, en una realización, para calcular el primer desfase para que esté basado en, y que sea preferiblemente igual a $(9 \times (q_0 - p_0) - 3 \times (q_1 - p_1) + 8) \gg 4$.

La Fig. 9 es un diagrama de bloques esquemático de una unidad de filtro de desbloqueo 200 de acuerdo con otra realización. La unidad de filtro de desbloqueo 200 comprende el calculador de primer desfase 210 y el modificador 220 de valor de píxel. Estas unidades 210, 220 operan como se ha explicado anteriormente con respecto a la Fig. 8. La unidad de filtro de desbloqueo 200 preferiblemente comprende también una primera unidad de acortamiento 230. Esta primera unidad de acortamiento 230 está configurada para acortar el primer desfase para que esté dentro de un intervalo de $-t_c$ y t_c . El valor de umbral t_c depende entonces de un bloqueo de QP asociado con el bloque y puede, por ejemplo, ser seleccionado de la Tabla 1 basándose en el valor de QP del bloque.

Una unidad de segundo acortamiento 240 opcional está implementada en la unidad de filtro de desbloqueo 200 para el acortamiento de los valores de píxel modificados calculados por el modificador de valor de píxel 220. Así, la unidad de segundo acortamiento 240 limita estos valores de píxel modificados para que estén dentro de un intervalo de cero y el valor máximo definido M.

En una realización preferida la unidad de filtro de desbloqueo 200 comprende también un calculador de segundo desfase 260 configurado para calcular un segundo desfase basado en $(p_2 + p_0 - 2 \times p_1 + 2 \times \Delta) / 4$ para la línea de píxeles en el bloque.

En una realización particular, el calculador de segundo desfase 260 está configurado para calcular el segundo desfase para que sea $g((p_2 + p_0 - 2 \times p_1 + 2 \times \Delta)/4)$, es decir, una función $g()$ o representación de $(p_2 + p_0 - 2 \times p_1 + 2 \times \Delta)/4$. Esta función preferiblemente da como resultado una representación entera de $(p_2 + p_0 - 2 \times p_1 + 2 \times \Delta)/4$ y preferiblemente la función es adecuada para implementación mediante hardware.

5 El segundo calculador de desfase 250 está configurado, en una realización, para calcular el segundo desfase para que esté basado en, sea preferiblemente igual a $((p_2 + p_0 + 1) \gg 1) - p_1 + \Delta \gg 1$.

La unidad de filtro de desbloqueo 200 preferiblemente comprende un calculador de tercer desfase 260 configurado para calcular un tercer desfase basado en $(q_2 + q_0 - 2 \times q_1 - 2 \times \Delta)/4$ para la línea de píxeles correspondiente en el bloque vecino.

10 En una realización particular, el calculador de tercer desfase 260 está configurado para calcular el tercer desfase para que sea $h((q_2 + q_0 - 2 \times q_1 - 2 \times \Delta)/4)$, es decir, una función $h()$ o representación de $(q_2 + q_0 - 2 \times q_1 - 2 \times \Delta)/4$. Esta función preferiblemente da como resultado una representación entera de $(q_2 + q_0 - 2 \times q_1 - 2 \times \Delta)/4$ y preferiblemente la función es adecuada para implementación mediante hardware. El calculador de tercer desfase 260 está configurado, en una realización, para calcular el tercer desfase para que esté

15 basado en, sea preferiblemente igual a $((q_2 + q_0 + 1) \gg 1) - q_1 - \Delta \gg 1$.

El modificador 220 de píxel de la unidad de filtro de desbloqueo 200 es entonces además configurado para modificar el valor de píxel del píxel siguiente más cercano a la frontera de bloque en la línea de píxeles en el bloque. El modificador 220 de píxel suma el segundo desfase calculado por el calculador de segundo desfase 250 al valor de píxel de este píxel. El modificador 220 de píxel está adicionalmente configurado para modificar el valor de píxel del

20 píxel siguiente más cercano a la frontera de bloque pero en la línea de píxeles correspondiente en el bloque vecino. Esta modificación se consigue sumando el primer desfase calculado por el calculador de tercer desfase 260 al valor de píxel de este píxel.

Una unidad de tercer acortamiento 270 opcional de la unidad de filtro de desbloqueo 200 está configurada para acortar el segundo desfase calculado por el calculador de segundo desfase 250 y el tercer desfase calculado por el

25 calculador de tercer desfase 260 para que esté dentro de un intervalo de $-t_{c2}$ y t_{c2} . El valor de umbral t_{c2} depende preferiblemente del valor de QP asociado con el bloque y se calcula ventajosamente sobre la base del valor de umbral t_c utilizado para el acortamiento del primer desfase. En una realización alternativa, la unidad de tercer acortamiento 270 se omite y cualquier acortamiento de los desfases segundo y tercero es llevado a cabo por el contrario por la unidad de primer acortamiento 230.

30 La unidad de segundo acortamiento 240 preferiblemente no solo acorta los valores de píxel modificados de los píxeles más cercanos a la frontera de bloque, sino también los valores de píxel modificados de los píxeles siguientes más cercanos a la frontera de bloque y calculados utilizando los desfases segundo y tercero, respectivamente. Así, también estos valores de píxel modificados están preferiblemente restringidos para que estén dentro del intervalo de cero al valor máximo definido M.

35 El calculador de segundo desfase 250 y el calculador de tercer desfase 260 podrían ser configurados para calcular los desfases segundo y tercero para cada línea de píxeles y cada línea de píxeles correspondiente para las cuales va a aplicarse el filtrado de desbloqueo sobre la frontera de bloque entre el bloque y el bloque vecino. En un planteamiento alternativo, la unidad de filtro de desbloqueo 200 lleva a cabo una selección acerca de si calcular solo el primer desfase y por ello modificar solo los valores de píxel de los píxeles más cercanos a la frontera de bloque o

40 calcular los desfases tanto primero como segundo y tercero y por ello modificar los valores de píxel de los píxeles más cercanos a y de los píxeles siguientes más cercanos a la frontera de bloque.

En una realización opcional, la unidad de filtro de desbloqueo 200 puede comprender un calculador de cuarto desfase 280 configurado para calcular un cuarto desfase para que esté basado en $(p_3 + p_1 - 2 \times p_2 + 2 \times \Delta_{p1})/4$.

En una realización particular, el calculador de cuarto desfase 280 está configurado para calcular el cuarto desfase

45 para que sea $b((p_3 + p_1 - 2 \times p_2 + 2 \times \Delta_{p1})/4)$, es decir, una función $b()$ o representación de $(p_3 + p_1 - 2 \times p_2 + 2 \times \Delta_{p1})/4$. Esta función preferiblemente proporciona una representación entera de $(p_3 + p_1 - 2 \times p_2 + 2 \times \Delta_{p1})/4$ y preferiblemente la función es adecuada para la implementación mediante

hardware. El calculador de cuarto desfase 280 está configurado, en una realización, para calcular el cuarto desfase para que esté basado en, sea preferiblemente igual a $\left(\left(\left(p_3 + p_1 + 1\right) \gg 1\right) - p_2 + \Delta_{p1}\right) \gg 1$.

La unidad de filtro de desbloqueo 200 podría comprender también un calculador de quinto desfase 290 opcional configurado para calcular un quinto desfase para que esté basado en $\left(q_3 + q_1 - 2 \times q_2 + 2 \times \Delta_{q1}\right) / 4$.

- 5 En una realización particular, el calculador de quinto desfase 290 está configurado para calcular el quinto desfase para que sea $k\left(\left(q_3 + q_1 - 2 \times q_2 + 2 \times \Delta_{q1}\right) / 4\right)$, es decir, una función $k(\)$ o representación de $\left(q_3 + q_1 - 2 \times q_2 + 2 \times \Delta_{q1}\right) / 4$. Esta función preferiblemente da como resultado una representación entera de $\left(q_3 + q_1 - 2 \times q_2 + 2 \times \Delta_{q1}\right) / 4$ y preferiblemente la función es adecuada para implementación mediante hardware. El calculador de quinto desfase 290 está configurado, en una realización, para calcular el quinto desfase para que
- 10 esté basado en, sea preferiblemente igual a $\left(\left(\left(q_3 + q_1 + 1\right) \gg 1\right) - q_2 + \Delta_{q1}\right) \gg 1$.

- En esta realización, el modificador de valor de píxel 220 está también configurado para modificar el valor de píxel del píxel segundo siguiente más cercano a la frontera de bloque en la línea de píxeles en el bloque, sumando el cuarto desfase calculado por el calculador de cuarto desfase 280 al valor de píxel de este píxel. El modificador de valor de píxel 220 adicionalmente modifica, en esta realización, el valor de píxel del píxel segundo siguiente más cercano a la
- 15 frontera de bloque en la línea de píxeles correspondiente en el bloque vecino, sumando el quinto desfase calculado por el calculador de quinto desfase al valor de píxel de este píxel.

- La unidad de segundo acortamiento 240 preferiblemente procesa los valores de píxel modificados para restringirlos dentro del intervalo de cero y el valor máximo predefinido M. La unidad de tercer acortamiento 270 puede utilizarse también para acortar los desfases cuarto y quinto de manera similar a los desfases segundo y tercero. Si no, la
- 20 unidad de filtro de desbloqueo 200 comprende una cuarta unidad de acortamiento configurada para acortar los desfases cuarto y quinto para que estén dentro de un intervalo que tiene puntos de extremo que están definidos sobre la base del valor de QP del bloque y preferiblemente basados en el valor de umbral t_c .

- Un aspecto particular se refiere a una unidad de filtro de desbloqueo para reducir las aberraciones de bloqueo en una frontera de bloque entre un bloque de múltiples píxeles y un bloque vecino de múltiples píxeles en un fotograma de video. Con referencia a la Fig. 8, la unidad de filtro de desbloqueo 100 comprende un calculador de primer
- 25 desfase 110 configurado para calcular un primer desfase basado en un valor de píxel de un píxel más cercano a la frontera de bloque en una línea de píxeles en el bloque, un valor de píxel de un píxel siguiente más cercano a la frontera de bloque en la línea de píxeles, un valor de píxel de un píxel más cercano a la frontera de bloque en una línea de píxeles correspondiente u opuesta en el bloque vecino y un valor de píxel de un píxel siguiente más cercano
- 30 a la frontera de bloque en la línea de píxeles correspondiente. La línea de píxeles y la línea de píxeles vecina son perpendiculares a la frontera de bloque. El primer desfase es calculado por el calculador de primer desfase 110 sobre la base de estos valores de píxel y de una fórmula que da como resultado un valor de desfase que se aproxima a cero, preferiblemente es igual a cero, cuando los valores de píxel crecen o decrecen linealmente o son iguales cuando se desplazan a lo largo de la línea de píxeles y de la línea de píxeles correspondiente y da como
- 35 resultado un valor de desfase que aproxima un escalón en los valores de píxel cuando los valores de píxel crecen y decrecen en escalones cuando se desplazan a lo largo de la línea de píxeles y de la línea de píxeles correspondiente. La unidad de filtro de desbloqueo 100 comprende también un modificador de valor de píxel 120 configurado para modificar el valor de píxel del píxel más cercano a la frontera de bloque en la línea de píxeles sumando el primer desfase al valor de píxel del píxel más cercano a la frontera de bloque en la línea de píxeles para
- 40 formar un valor de píxel modificado. El modificador de valor de píxel 120 está además configurado para modificar el valor de píxel del píxel más cercano a la frontera de bloque en la línea de píxeles correspondiente restando el primer desfase del valor de píxel del píxel más cercano a la frontera de bloque en la línea de píxeles correspondiente para formar un valor de píxel modificado.

- Aunque la unidad respectiva 110, 120 y 210 – 290 descrita con respecto a las Figs. 8 y 9 se ha descrito como
- 45 unidades físicamente separadas 110, 120, y 210 – 290 en el dispositivo 100, 200, y todas pueden ser circuitos de propósito especial, tal como ASICs (Circuitos integrados específicos para una aplicación – Application Specific Integrated Circuits, en inglés), son posibles realizaciones alternativas del dispositivo 100, 200, donde algunas o todas las unidades 110, 120, 210 – 290 están implementadas como módulos de programa informático que se ejecutan en un procesador de propósito general. Tal realización se describe en la Fig. 10.

- La Fig. 10 ilustra esquemáticamente una realización de un ordenador 70 que tiene una unidad de procesamiento 72, tal como un DSP (Procesador de Señal Digital – Digital Signal Processor, en inglés) o CPU (Unidad de
- 50 Procesamiento Central – Central Processing Unit, en inglés). La unidad de procesamiento 72 puede ser una sola unidad o una pluralidad de unidades para ejecutar diferentes etapas del método descrito en esta memoria. El ordenador 70 comprende también una unidad de entrada / salida (I/O – Input / Output, en inglés) 71 para la

recepción de fotogramas de video grabados o generados o fotogramas de video codificados y fotogramas de video codificados o datos de video descodificados. La unidad de I/O 71 ha sido ilustrada como una sola unidad en la Fig. 10, pero puede asimismo ser en forma de una unidad de entrada separada y una unidad de salida separada.

5 Además, el ordenador 70 comprende al menos un producto de programa informático 73 en forma de una memoria no volátil, por ejemplo una EEPROM (Memoria de solo lectura programable borrable eléctricamente – Electrically Erasable Programmable Read-Only Memory, en inglés), una memoria rápida o una unidad de disco. El producto de programa informático 73 comprende un programa informático 74, que comprende un medio de código que cuando es ejecutado en el ordenador 70, tal como mediante la unidad de procesamiento 72, hace que el ordenador 70 ejecute las etapas del método descritas anteriormente con respecto a la Fig. 1. Por ello, en una realización el medio de código en el programa informático 74 comprende un módulo de cálculo de primer desfase 310 para calcular el primer desfase y un módulo de modificación de primer desfase o un modificador de valor de píxel 320 para modificar los valores de píxel de un módulo de filtro de desbloqueo 300 o un dispositivo de filtro de desbloqueo. Estos módulos 310, 320 esencialmente ejecutan las etapas del diagrama de flujo de la Fig. 1 cuando se ejecutan en la unidad de procesamiento 72. Así, cuando los diferentes módulos 310, 320 son ejecutados en la unidad de procesamiento 72 corresponden a las unidades correspondientes 110, 120 de la Fig. 8 y a las unidades 210, 220 de la Fig. 9.

El programa informático 74 puede adicionalmente comprender un módulo de primer acortamiento, un módulo de segundo acortamiento, un módulo de cálculo de segundo desfase, un módulo de cálculo de tercer desfase y opcionalmente también un módulo de tercer acortamiento, un módulo de cálculo de cuarto desfase y un módulo de cálculo de quinto desfase para llevar a cabo la operación de las correspondientes unidades 230 – 290 en la Fig. 9.

20 El ordenador 70 de la Fig. 10 puede ser un equipo de usuario o estar incluido en un equipo de usuario 80. En tal caso, el equipo de usuario 80 puede adicionalmente comprender o estar conectado a una pantalla 88 para mostrar los datos de video.

La unidad de filtro de desbloqueo 100, 200 de las Figs. 8 y 9 se utilizan preferiblemente en la codificación de video. Funciona y está por lo tanto preferiblemente implementada tanto en un codificador de video como en un descodificador de video. El codificador de video puede ser implementado preferiblemente en hardware, pero también en software. Lo mismo vale para el codificador de video.

La Fig. 11 es un diagrama de bloques esquemático de un codificador 40 para decodificar un bloque de píxeles en un fotograma de video de una secuencia de video de acuerdo con una realización.

30 Un bloque de píxeles actual se predice realizando una estimación de movimiento mediante un estimador de movimiento 50 de un bloque de píxeles ya proporcionado en el mismo fotograma o en un fotograma previo. 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 de movimiento 50 para proporcionar una inter predicción de bloque de píxeles.

35 Un intra indicador 49 calcula una intra predicción de los bloques de píxeles actuales. Las salidas del estimador / compensador de movimiento 50 y el intra indicador 49 son introducidas en un selector 51 que bien selecciona intra o inter predicción para los bloques de píxeles actuales. La salida del selector 51 es introducida en un calculador de error en forma de un sumador 41 que también recibe los valores de píxel del bloque de píxeles actual. El sumador 41 calcula y proporciona un error residual como la diferencia en los valores de píxel entre el bloque de píxeles y su predicción.

40 El error se transforma en un transformador 42, tal como mediante una transformada de coseno discreta, y cuantificado mediante un cuantificador 43 seguido por una codificación en un codificador 44, tal como mediante un codificador de entropía. En inter codificación, también el vector de movimiento estimado es introducido en el codificador 44 para generar la representación codificada del bloque de píxeles actual.

45 El error residual transformado y cuantificado para el bloque de píxeles actual se proporciona también a un cuantificador inverso 45 y a un transformador inverso 46 para obtener el error residual original. Este error es sumado por un sumador 47 a la salida de predicción del bloque desde el compensador de movimiento 50 o el intra indicador 49 para crear un bloque de píxeles de referencia que pueda ser utilizado en la predicción y la codificación de un bloque de píxeles siguiente. Este nuevo bloque de referencia es procesado primero por una unidad de filtro de desbloqueo 100 de acuerdo con las realizaciones para realizar un filtrado de desbloqueo para combatir todas las aberraciones de bloqueo. El nuevo bloque de referencia procesado es entonces temporalmente almacenado en una memoria temporal de fotogramas 48, donde está disponible para el intra indicador 49 y para el estimador / compensador de movimiento 50.

55 La Fig. 12 es un diagramas de bloques esquemático de un descodificador 60 que comprende una unidad de filtro de desbloqueo 100 de acuerdo con las realizaciones. El descodificador 60 comprende un descodificador 61, tal como un descodificador en entropía, para la descodificación de una representación codificada de un bloque de píxeles para obtener un conjunto de errores residuales cuantificados y transformados. Estos errores residuales son descuantificados en un cuantificador inverso 62 e inversamente transformados por un transformador inverso 63 para obtener un conjunto de errores residuales.

Estos errores residuales son sumados en un sumador 64 a los valores de píxel de un bloque de píxeles de referencia. El bloque de referencia es determinado mediante un estimador / compensador de movimiento 67 o un intra indicador 66, dependiendo de si se lleva a cabo una inter o una intra predicción. Un selector 68 es por ello interconectado al sumador 64 y al estimador / compensador de movimiento 67 y al intra indicador 66. La salida del bloque de píxeles descodificada resultante del sumador 64 es introducida en una unidad de filtro de desbloqueo 100 de acuerdo con las realizaciones para filtrar para desbloqueo todas las aberraciones de bloqueo. El bloque de píxeles, filtrado es extraído del descodificador 60 y es además preferiblemente temporalmente proporcionado a una memoria temporal de fotogramas 65 y puede ser utilizado como bloque de píxeles de referencia para un bloque de píxeles siguiente para ser descodificado. La memoria temporal de fotogramas 65 está por ello conectada al estimador / compensador de movimiento 67 para que los bloques de píxeles almacenados estén disponibles para el estimador / compensador de movimiento 67.

La salida del sumador 64 es preferiblemente introducida también en el intra indicador 66 para ser utilizada como un bloque de píxeles de referencia no filtrado.

En las realizaciones descritas en las Figs. 11 y 12 la unidad de filtro de desbloqueo 100 lleva a cabo un filtrado de desbloqueo como el llamado filtrado en bucle. En una implementación alternativa en el descodificador 60 la unidad de filtro de desbloqueo 100 está dispuesta para realizar el llamado filtrado de posprocesamiento. En tal caso, la unidad de filtro de desbloqueo 100 opera sobre los fotogramas de salida fuera del bucle formado por el sumador 64, la memoria temporal de fotogramas 65, el intra indicador 66, el estimador / compensador de movimiento 67 y el selector 68. No se realiza entonces típicamente ningún filtrado de desbloqueo en el codificador.

La Fig. 13 es un diagrama de bloques esquemático de un equipo de usuario o terminal de medios 80 que contiene un descodificador 60 con una unidad de filtro de desbloqueo. El equipo de usuario 80 puede ser cualquier dispositivo que tenga funciones de descodificación de medios que opera sobre un flujo de video codificado de fotogramas de video descodificados para descodificar con ello los fotogramas de video y proporcionar los datos de video. Ejemplos no limitativos de tales dispositivos incluyen teléfonos móviles y otros reproductores de medios portátiles, tabletas, ordenadores, blocs de notas, grabadores de video personales, reproductores de multimedios, servidores de video en tiempo real, descodificadores de televisión, TVs, ordenadores, descodificadores, consolas de juegos, etc. El equipo de usuario 80 comprende una memoria 84 configurada para almacenar fotogramas de video codificados. Estos fotogramas de video codificados pueden haber sido generados por el propio equipo de usuario 80. En tal caso, el equipo de usuario 80 preferiblemente comprende un motor de medios o un grabador junto con un codificador conectado, tal como el codificador de la Fig. 11. Si no, los fotogramas de video codificados son generados por algún otro dispositivo y transmitidos de manera inalámbrica o transmitidos por cable al equipo de usuario 80. El equipo de usuario 80 comprende entonces un transmisor receptor (transmisor y receptor) o un puerto de entrada y salida 82 para conseguir la transferencia de datos.

Los fotogramas de video codificados son traídos desde la memoria 84 a un descodificador 60, tal como el descodificador ilustrado en la Fig. 12. El descodificador 60 comprende una unidad de filtro de desbloqueo 100 de acuerdo con las realizaciones. El descodificador 60, a continuación, descodifica los fotogramas de video codificados a fotogramas de video descodificados. Los fotogramas de video descodificados son proporcionados a un reproductor de medios 86 que está configurado para traducir los fotogramas de video descodificados a datos de video que son visualizables en un visualizador, pantalla 88 de o conectado al equipo de usuario 80.

En la Fig. 13, el equipo de usuario 80 ha sido ilustrado comprendiendo tanto el descodificador 60 como el reproductor de medios 86, estando el descodificador 60 implementado como parte del reproductor de medios 86. Esto debería, no obstante, considerarse meramente como un ejemplo ilustrativo pero no limitativo de una realización de implementación para el equipo de usuario 80. También son posibles implementaciones distribuidas cuando el descodificador 60 y el reproductor de medios 86 son proporcionados en dos dispositivos separados físicamente y dentro del alcance del equipo de usuario 80, tal como se utiliza en esta memoria. La pantalla 88 podría proporcionarse también como un dispositivo separado conectado al equipo de usuario 80, cuando se está produciendo el procesamiento de los datos actuales.

Como se ilustra en la Fig. 14, el codificador 40 y/o el descodificador 60, tal como se ilustra en las Figs. 11 y 12, pueden ser implementados en un dispositivo de red 30 que está o que pertenece a un nodo de red en una red de comunicación 32 entre una unidad de transmisión 34 y un equipo de usuario de recepción 36. Tal dispositivo de red 30 puede ser un dispositivo para convertir video de acuerdo con un estándar de codificación de video en otro estándar de codificación de video, por ejemplo, si se ha establecido que el equipo de usuario de recepción 36 solo es capaz de o prefiere otro estándar de codificación de video distinto del enviado desde la unidad de transmisión 34. El dispositivo de red 30 puede ser en forma de o estar comprendido en una estación de base de radio, un Nodo B o en cualquier otro nodo de red en una red de comunicación 32, tal como una red de radio.

Las realizaciones descritas anteriormente deben ser entendidas como ejemplos ilustrativos de la presente invención. Resultará evidente para los expertos en la materia que pueden realizarse varias modificaciones, combinaciones y cambios a las realizaciones sin separarse del alcance de la presente invención. En particular, pueden combinarse diferentes soluciones parciales en las diferentes realizaciones en otras configuraciones, cuando sea técnicamente posible. El alcance de la presente invención está, no obstante, definido por las reivindicaciones adjuntas.

REIVINDICACIONES

1. Un método de reducir las aberraciones de bloqueo en una frontera de bloque (1) entre un bloque (10) de múltiples píxeles (11, 13, 15, 17) y un bloque vecino (20) de múltiples píxeles (21, 23, 25, 27) en un fotograma de video, comprendiendo el citado método las etapas de:

5 calcular (S1) un primer desfase;

acortar (S11) el citado primer desfase para que esté dentro de un intervalo de $-t_c$ y t_c , ajustando el citado primer desfase para que sea igual a $-t_c$ si el citado primer desfase es menor que $-t_c$ y ajustando el citado primer desfase para que sea igual a t_c si el citado primer desfase es mayor que t_c , donde t_c es un valor de umbral que depende de un valor de parámetro de cuantificación asignado al citado bloque (10);

10 modificar (S2) un valor de píxel p_0 de un píxel (11) más cercano a la citada frontera de bloque (1) en una línea (12) de píxeles (11, 13, 15, 17) en el citado bloque (10), donde la citada línea (12) de píxeles (11, 13, 15, 17) es perpendicular a la citada frontera de bloque (1), sumando el citado primer desfase al citado valor de píxel p_0 del citado píxel (11) más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) para formar un valor de píxel modificado p_0' del citado píxel (11) más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17); y

15 modificar (S3) un valor de píxel q_0 de un píxel (21) más cercano a la citada frontera de bloque (1) en una línea (22) correspondiente de píxeles (21, 23, 25, 27) en el citado bloque vecino (20), donde la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) es perpendicular a la citada frontera de bloque (1), restando el citado primer desfase del citado valor de píxel q_0 del citado píxel (21) más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) para formar un valor de píxel modificado q_0' del citado píxel (21) más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27), y donde el método está **caracterizado por**

20 que el primer desfase se calcula para que sea igual a $(9 \times (q_0 - p_0) - 3 \times (q_1 - p_1) + 8) \gg 4$, donde \gg denota una operación de desplazamiento a la derecha, p_1 denota un valor de píxel de un píxel (13) siguiente más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) y q_1 denota un valor de píxel de un píxel (23) siguiente más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27).

2. El método de acuerdo con la reivindicación 1, que comprende además:

30 acortar (S21) el citado valor de píxel modificado p_0' del citado píxel (11) más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) para que esté en un intervalo de cero y un valor máximo definido ajustando el citado valor de píxel modificado p_0' del citado píxel (11) más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) para que sea igual a cero si el citado valor de píxel modificado p_0' del citado píxel (11) más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) es menor que cero, y ajustando el citado valor de píxel modificado p_0' del citado píxel (11) más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) para que sea igual al citado valor máximo definido si el citado valor de píxel modificado p_0' del citado píxel (11) más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) es mayor que el citado valor máximo definido; y

40 acortar (S21) el citado valor de píxel modificado q_0' del citado píxel (21) más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) para que esté dentro de un intervalo de cero y el citado valor máximo definido ajustando el citado valor de píxel modificado q_0' del citado píxel (21) más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) para que sea igual a cero si el citado valor de píxel modificado q_0' del citado píxel (21) más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) es menor que cero, y ajustando el citado valor de píxel modificado q_0' del citado píxel (21) más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) para que sea igual al citado valor máximo definido si el citado valor de píxel modificado q_0' del citado píxel (21) más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) es mayor que el citado valor máximo definido.

3. El método de acuerdo con cualquiera de las reivindicaciones 1 o 2, que comprende además:

50 calcular (S30) un segundo desfase para que sea igual a $((p_2 + p_0 + 1) \gg 1) - p_1 + \Delta \gg 1$, donde p_2 denota un valor de píxel de un píxel (15) segundo siguiente más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) y Δ denota el citado primer desfase;

calcular (S31) un tercer desfase para que sea

igual a $\left(\left(\left(q_2 + q_0 + 1\right) \gg 1\right) - q_1 - \Delta\right) \gg 1$,

donde q_2 denota un valor de píxel de un píxel (25) segundo siguiente más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27);

5 modificar (S32) el citado valor de píxel p_1 del citado píxel (13) siguiente más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) sumando el citado segundo desfase al citado valor de píxel p_1 del citado píxel (13) siguiente más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) para formar un valor de píxel modificado p_1' del citado píxel (13) siguiente más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17); y

10 modificar (S33) el citado valor de píxel q_1 del citado píxel (23) siguiente más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) sumando el citado tercer desfase al citado valor de píxel q_1 del citado píxel (23) siguiente más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) para formar un valor de píxel modificado q_1' del citado píxel (23) siguiente más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27).

15 4. Una unidad de filtro de desbloqueo (100, 200) para reducir aberraciones de bloqueo en una frontera de bloque (1) entre un bloque (10) de múltiples píxeles (11, 13, 15, 17) y un bloque vecino (20) de múltiples píxeles (21, 23, 25, 27) en un fotograma de video, comprendiendo la citada unidad de filtro de desbloqueo (100, 200):

un calculador de primer desfase (110, 210) configurado para calcular un primer desfase;

20 una unidad de primer acortamiento (230) configurada para acortar el citado primer desfase dentro de un intervalo de $-t_c$ y t_c ajustando el citado primer desfase para que sea igual a $-t_c$ si el citado primer desfase es menor que $-t_c$ y ajustando el citado primer desfase para que sea igual a t_c si el citado primer desfase es mayor que t_c , donde t_c es un valor de umbral que depende del valor de un parámetro de cuantificación asignado al citado bloque (10), y

25 un modificador de valor de píxel (120, 220) configurado para modificar un valor de píxel p_0 de un píxel (11) más cercano a la citada frontera de bloque (1) en una línea (12) de píxeles (11, 13, 15, 17) en el citado bloque (10), donde la citada línea (12) de píxeles (11, 13, 15, 17) es perpendicular a la citada frontera de bloque (1), sumando el citado primer desfase al citado valor de píxel p_0 del citado píxel (11) más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) para formar un valor de píxel modificado p_0' del citado píxel (11) más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) y para modificar un valor de píxel q_0 de un píxel (21) más cercano a la citada frontera de bloque (1) en una línea (22) correspondiente de píxeles (21, 23, 25, 27) en el citado bloque vecino (20), donde la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) es perpendicular a la citada frontera de bloque (1), restando el citado primer desfase del citado valor de píxel q_0 del citado píxel (21) más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) para formar un valor de píxel modificado q_0' del citado píxel (21) más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27), y

35 está **caracterizada por**

40 que el primer desfase se calcula para que sea igual a $(9 \times (q_0 - p_0) - 3 \times (q_1 - p_1) + 8) \gg 4$, donde \gg denota una operación de desplazamiento a la derecha, p_1 denota un valor de píxel de un píxel (13) siguiente más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) y q_1 denota un valor de píxel de un píxel (23) siguiente más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27).

5. La unidad de filtro de desbloqueo de acuerdo con la reivindicación 4, que comprende además una unidad de segundo acortamiento (240) configurada para acortar el citado valor de píxel modificado p_0' del citado píxel (11) más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) para que esté dentro de un intervalo de cero y un valor máximo definido, ajustando el citado valor de píxel modificado p_0' del citado píxel (11) más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) para que sea igual a cero si el citado valor de píxel modificado p_0' del citado píxel (11) más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) es menor que cero y ajustando el citado valor de píxel modificado p_0' del citado píxel (11) más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) para que sea igual al citado valor máximo definido si el citado valor de píxel modificado p_0' del citado píxel (11) más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) es mayor que el citado valor máximo definido, y para acortar el citado valor de píxel modificado q_0' del citado píxel (21) más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) para que esté dentro de un intervalo de cero y el citado valor máximo definido, ajustando el citado valor de píxel modificado q_0' del citado píxel (21) más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) para que sea igual a cero si el citado valor de píxel modificado q_0' del citado píxel (21) más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) es

menor que cero, y ajustando el citado valor de píxel modificado q_0' del citado píxel (21) más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) para que sea igual al citado valor máximo definido si el citado valor de píxel modificado q_0' del citado píxel (21) más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) es mayor que el citado valor máximo definido.

5 6. La unidad de filtro de desbloqueo de acuerdo con cualquiera de las reivindicaciones 4 o 5, que comprende además:

un calculador de segundo desfase (250) configurado para calcular un segundo desfase para que sea igual a $\left(\left(\left(p_2 + p_0 + 1\right) \gg 1\right) - p_1 + \Delta\right) \gg 1$, donde p_2 denota un valor de píxel de un píxel (15) segundo siguiente más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) y Δ denota el citado primer desfase; y

10

un calculador de tercer desfase (260) configurado para calcular un tercer desfase para que sea igual a $\left(\left(\left(q_2 + q_0 + 1\right) \gg 1\right) - q_1 - \Delta\right) \gg 1$,

15 donde q_2 denota un valor de píxel de un píxel (25) segundo siguiente más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27),

en la que el citado modificador de valor de píxel (220) está configurado para modificar el citado valor de píxel p_2 del citado píxel (15) segundo siguiente más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) añadiendo el citado segundo desfase al citado valor de píxel p_2 del citado píxel (15) segundo siguiente más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) para formar un valor de píxel modificado p_2' del citado píxel (15) segundo siguiente más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) y para modificar el citado valor de píxel q_2 del citado píxel (25) segundo siguiente más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) sumando el citado tercer desfase al citado valor de píxel q_2 del citado píxel (25) segundo siguiente más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) para formar un valor de píxel modificado q_2' del citado píxel (25) segundo siguiente más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27).

20

25

7. La unidad de filtro de desbloqueo de acuerdo con la reivindicación 6, que comprende además una unidad de tercer acortamiento (270) configurada para acortar el citado segundo desfase para que esté dentro de un intervalo de $-t_{c2}$ y t_{c2} ajustando el citado segundo desfase para que sea igual a $-t_{c2}$ si el citado segundo desfase es menor que $-t_{c2}$ y ajustando el citado segundo desfase para que sea igual a t_{c2} si el segundo desfase es mayor que t_{c2} , donde t_{c2} es un valor de umbral que depende de un valor del parámetro de cuantificación asignado al citado bloque, y para acortar el citado tercer desfase dentro de un intervalo de $-t_{c2}$ y t_{c2} ajustando el citado tercer desfase para que sea igual a $-t_{c2}$ si el citado tercer desfase es menor que $-t_{c2}$ y ajustando el citado tercer desfase para que sea igual a t_{c2} si el citado tercer desfase es mayor que t_{c2} .

30

35 8. La unidad de filtro de desbloqueo de acuerdo con cualquiera de las reivindicaciones 6 y 7, que comprende además:

un calculador de cuarto desfase (280) configurado para calcular un cuarto desfase para que sea igual a $\left(\left(\left(p_3 + p_1 + 1\right) \gg 1\right) - p_2 + \Delta_{p1}\right) \gg 1$, donde p_3 denota un valor de píxel de un píxel (17) tercero siguiente más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) y Δ_{p1} denota el citado segundo desfase; y

40

un calculador de quinto desfase (290) configurado para calcular un quinto desfase para que sea igual a $\left(\left(\left(q_3 + q_1 + 1\right) \gg 1\right) - q_2 + \Delta_{q1}\right) \gg 1$, donde q_3 denota un valor de píxel de un tercer píxel (27) más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) y Δ_{q1} denota el citado tercer desfase,

45 en la que el citado modificador de valor de píxel (220) está configurado para modificar el citado valor de píxel p_2 del citado píxel (15) segundo siguiente más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) sumando el citado cuarto desfase al citado valor de píxel p_2 del citado píxel (15) segundo siguiente más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) para formar un valor de píxel modificado p_2' del citado píxel (15) segundo siguiente más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17), y para modificar el citado valor de píxel q_2 del citado píxel (25) segundo siguiente más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) sumando el citado quinto desfase al citado valor de píxel q_2 del citado píxel (25) segundo siguiente más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) para

50

formar un valor de píxel modificado q_2' del citado píxel (25) segundo siguiente más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27).

9. Un codificador (40) que comprende una unidad de filtro de desbloqueo (100) de acuerdo con cualquiera de las reivindicaciones 4 a 8.
- 5 10. Un decodificador (60) que comprende una unidad de filtro de desbloqueo (100) de acuerdo con cualquiera de las reivindicaciones 4 a 8.
11. Un equipo de usuario (80) que comprende:
 una memoria (84) configurada para almacenar fotogramas de video codificados; y que comprende
 un decodificador (60) de acuerdo con la reivindicación 10 configurado para decodificar los citados fotogramas de video codificados en fotograma de video descodificados; y
 un reproductor de medios (86) configurado para traducir los citados fotogramas de video descodificados a datos de video visualizables en una pantalla (88).
12. Un dispositivo de red (30) que se encuentra en o pertenece a un nodo de red en una red de comunicación (32) entre una unidad de transmisión (34) y un equipo de usuario de recepción (36), comprendiendo el citado dispositivo de red (30) un codificador (40) de acuerdo con la reivindicación 9 y/o un decodificador (60) de acuerdo con la reivindicación 10.
13. Un programa informático (74) para reducir las aberraciones de bloqueo en una frontera de bloque (1) entre un bloque (10) de múltiples píxeles (11, 13, 15, 17) y un bloque vecino (20) de múltiples píxeles (21, 23, 25, 27) en un fotograma de video, comprendiendo el citado programa informático (74) un medio que cuando es ejecutado en un ordenador (70) hace que el ordenador (70):
 calcule un primer desfase;
 acorte (S11) el citado primer desfase para que esté dentro de un intervalo $-t_c$ y t_c ajustando el citado primer desfase para que sea igual a $-t_c$ si el citado primer desfase es menor que $-t_c$ y ajustando el citado primer desfase para que sea igual a t_c si el citado primer desfase es mayor que t_c , donde t_c es un valor de umbral que depende de un valor del parámetro de cuantificación asignado al citado bloque (10);
 modifique un valor de píxel p_0 de un píxel (11) más cercano a la citada frontera de bloque (1) en una línea (12) de píxeles (11, 13, 15, 17) en el citado bloque (10), donde la citada línea (12) de píxeles (11, 13, 15, 17) es perpendicular a la citada frontera de bloque (1), sumando el citado primer desfase al citado valor de píxel del citado píxel (11) más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17); y
 modifique un valor de píxel q_0 de un píxel (21) más cercano a la citada frontera de bloque (1) en una línea (22) correspondiente de píxeles (21, 23, 25, 27) en el citado bloque vecino (20), donde la citada línea (22) correspondiente de píxeles (21, 23, 25, 27) es perpendicular a la citada frontera de bloque (1), restando el primer desfase del citado valor de píxel del citado píxel (21) más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27), y
caracterizado por un medio de código que cuando se ejecuta en el ordenador (70) hace que el ordenador (70) calcule el primer desfase para que sea igual a $(9 \times (q_0 - p_0) - 3 \times (q_1 - p_1) + 8) \gg 4$, donde \gg denota una operación de desplazamiento a la derecha, p_1 denota un valor de píxel de un píxel (13) siguiente más cercano a la citada frontera de bloque (1) en la citada línea (12) de píxeles (11, 13, 15, 17) y q_1 denota un valor de píxel de un píxel (23) siguiente más cercano a la citada frontera de bloque (1) en la citada línea (22) correspondiente de píxeles (21, 23, 25, 27).

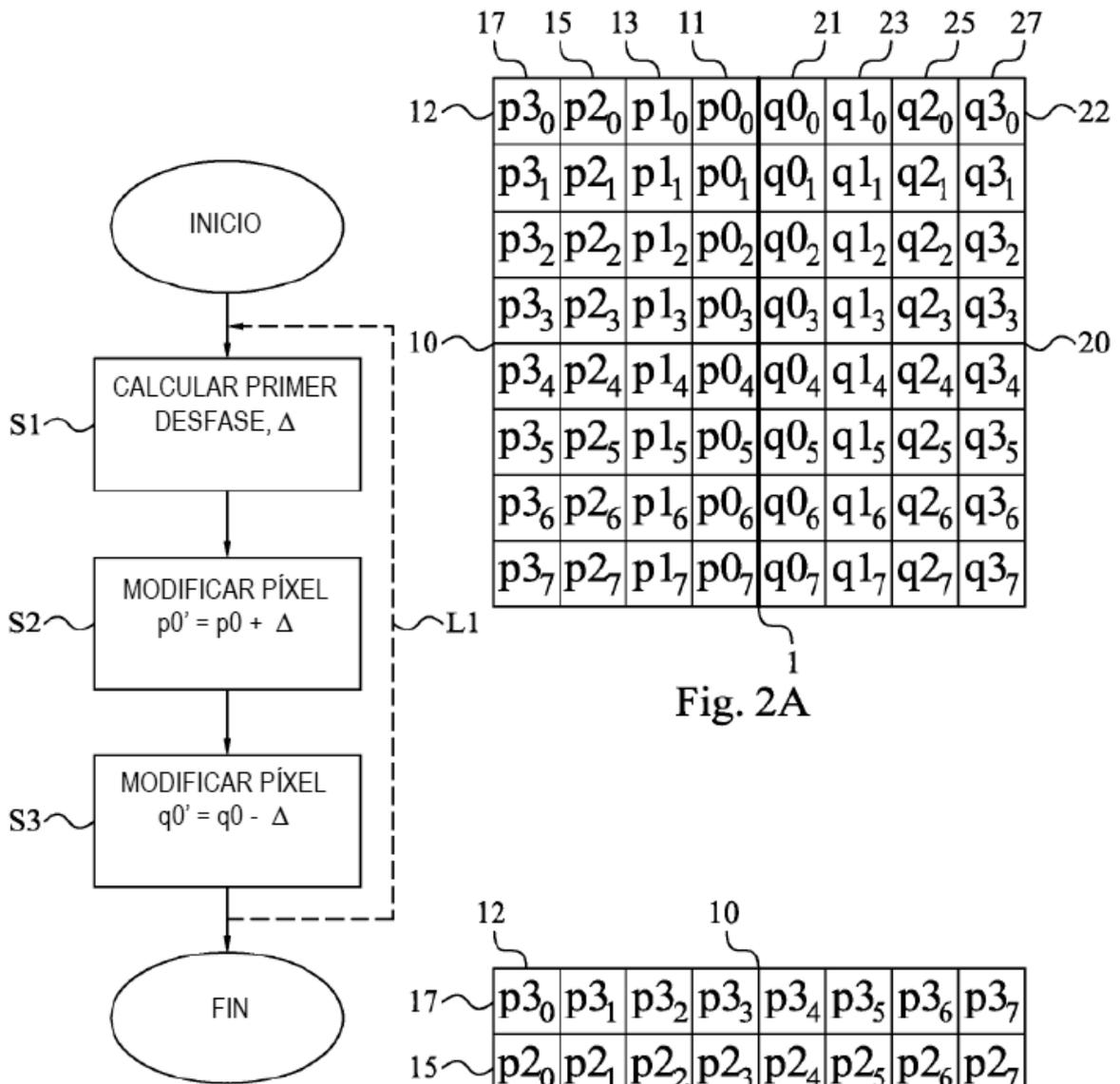


Fig. 2A

Fig. 1

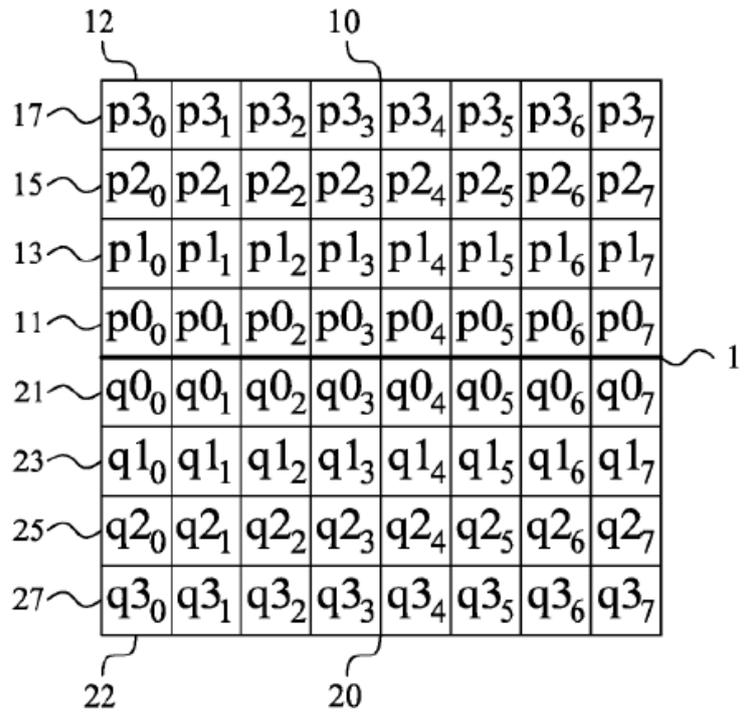


Fig. 2B

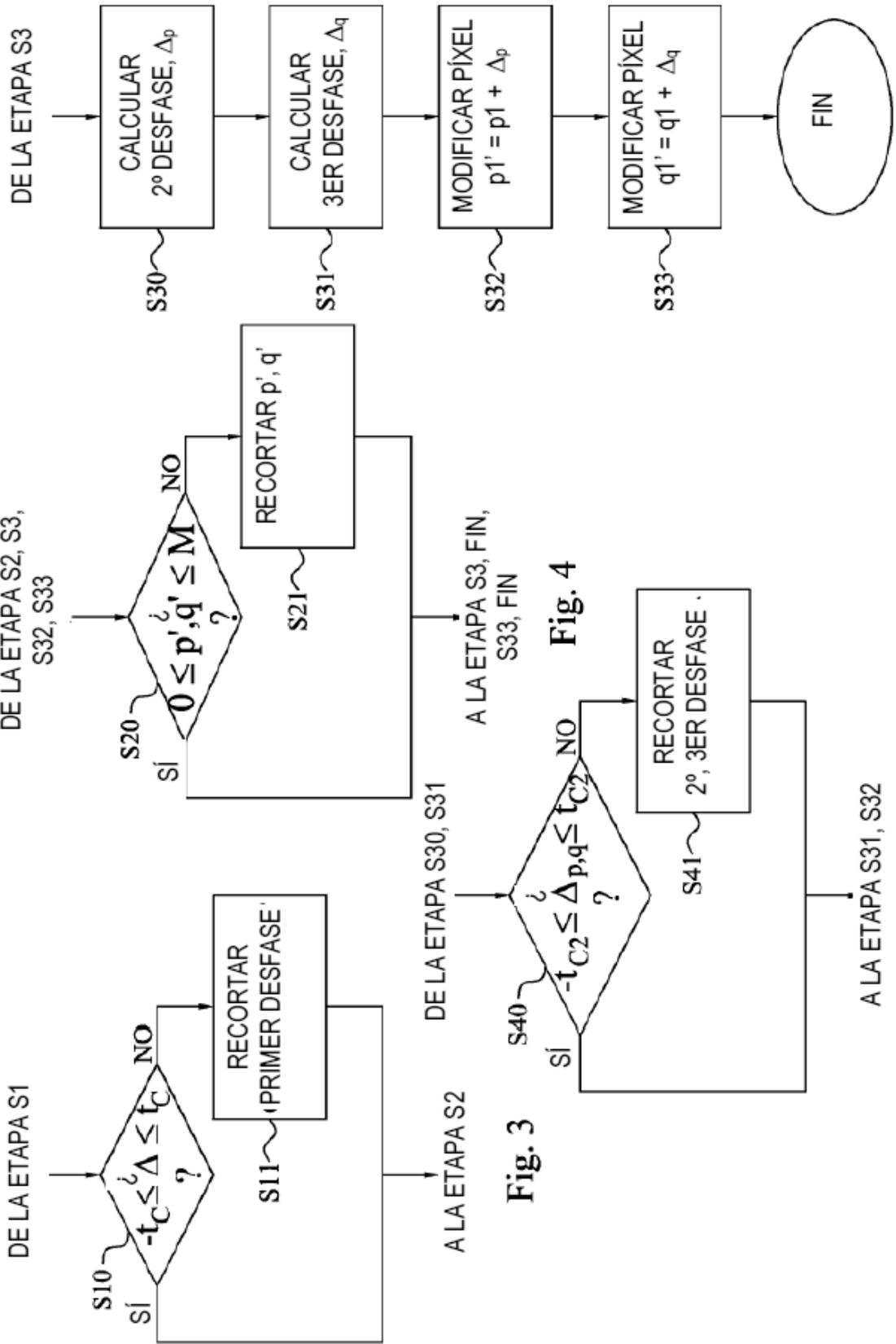


Fig. 5

Fig. 6

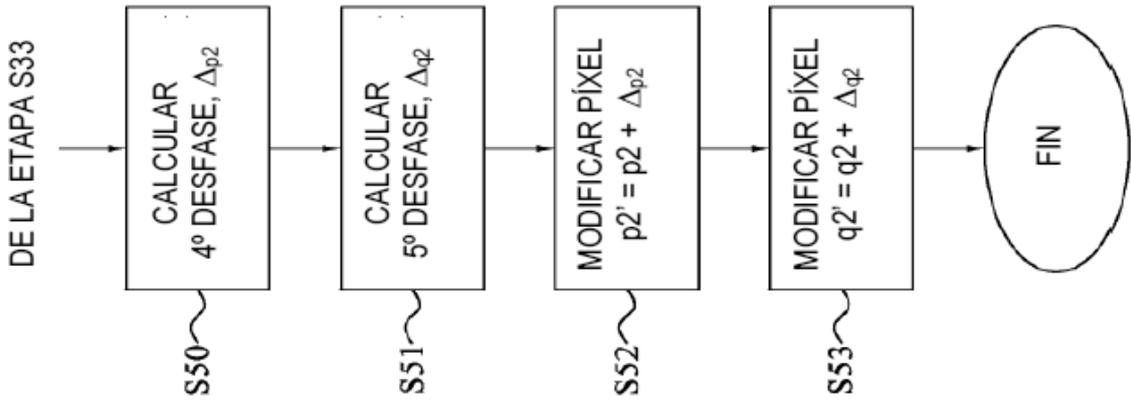


Fig. 7

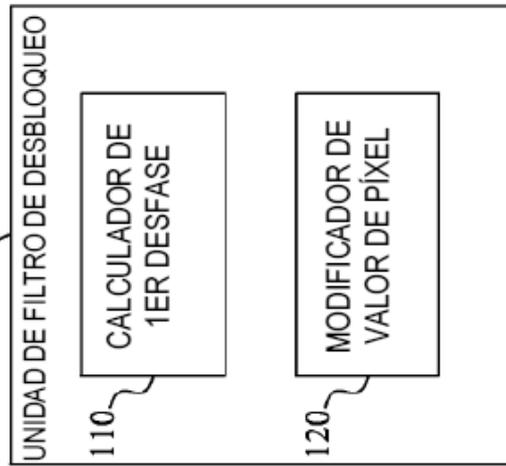


Fig. 8

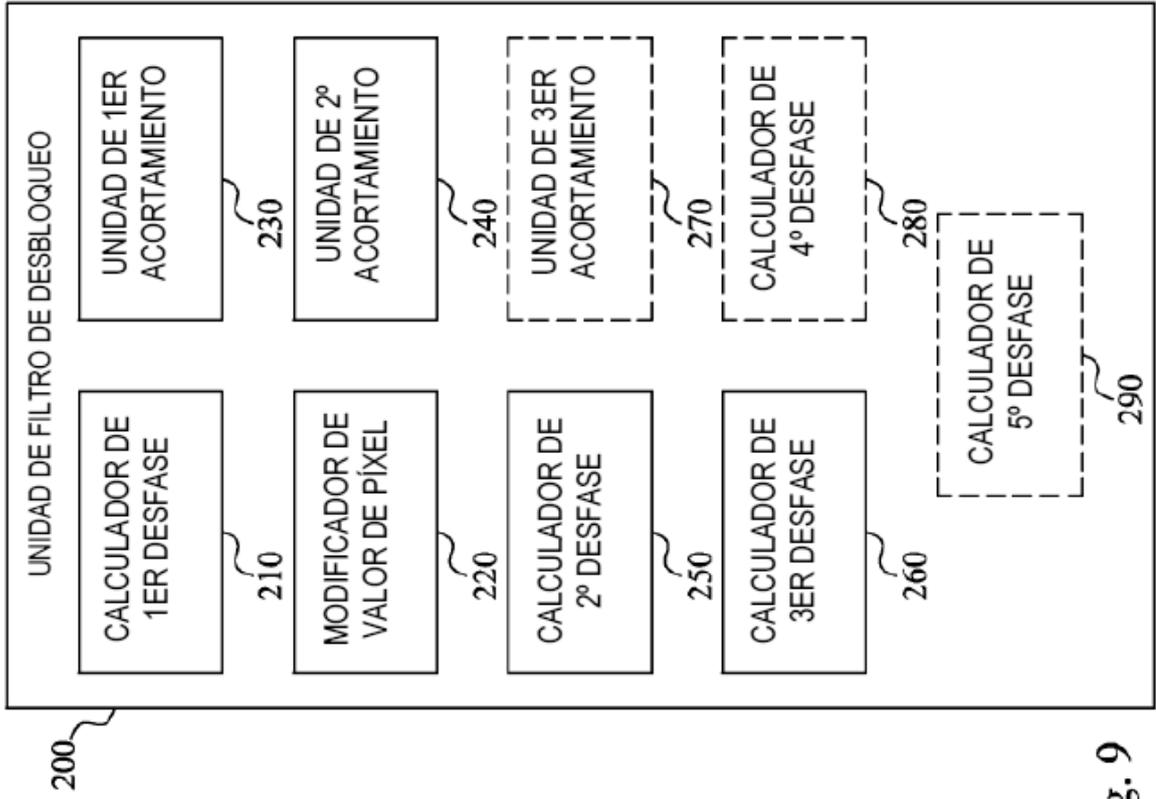


Fig. 9

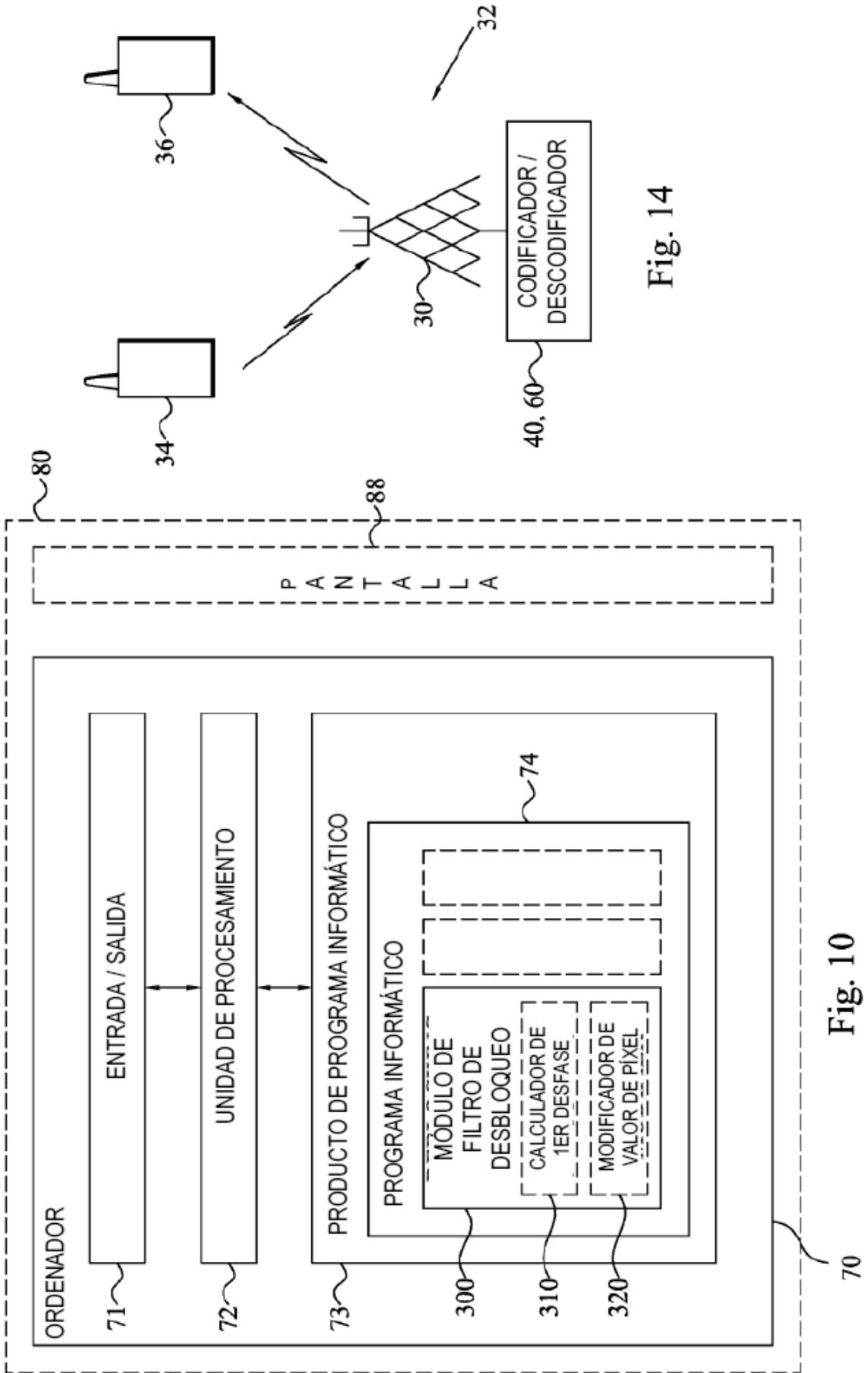


Fig. 14

Fig. 10

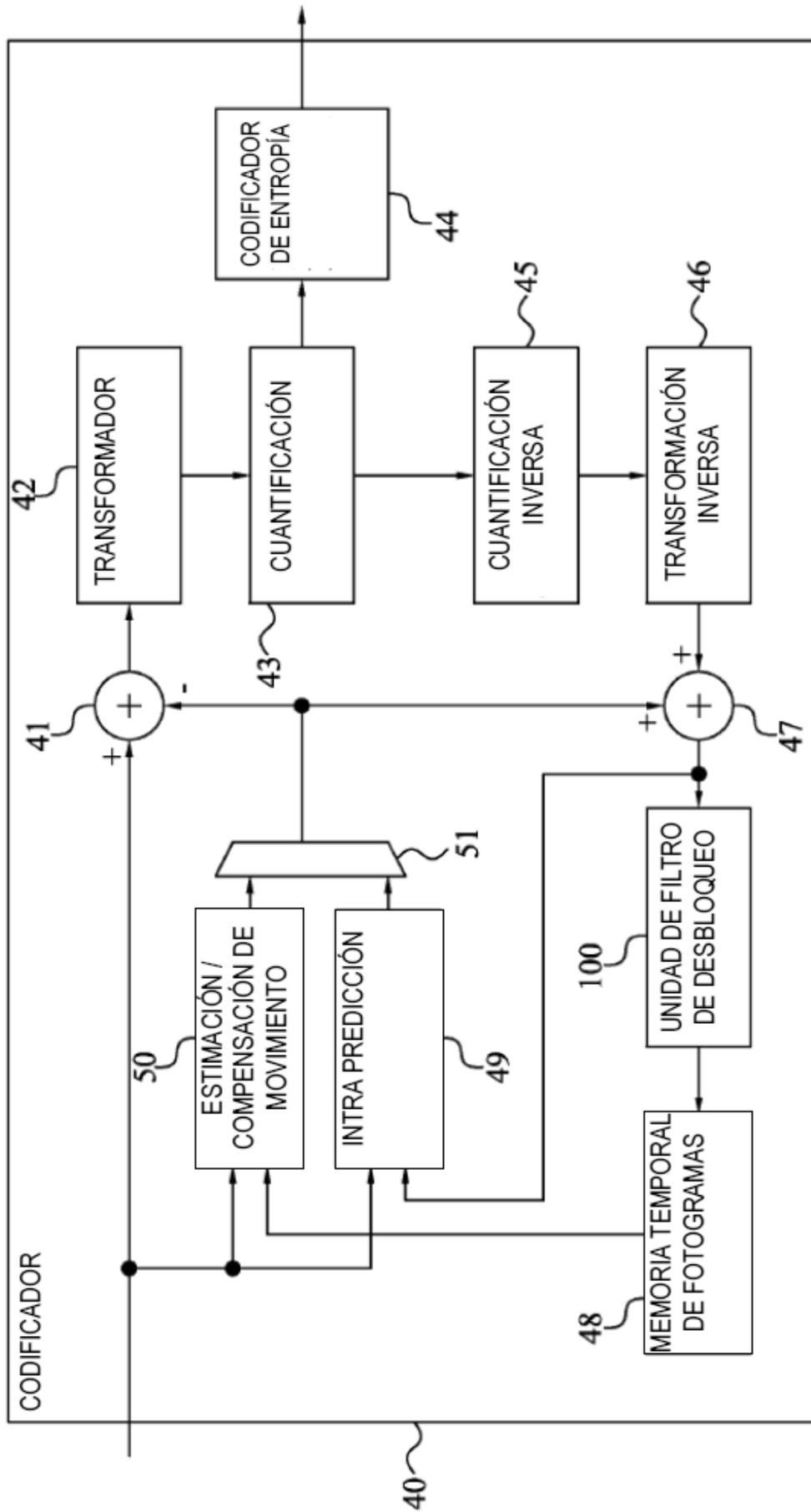


Fig. 11

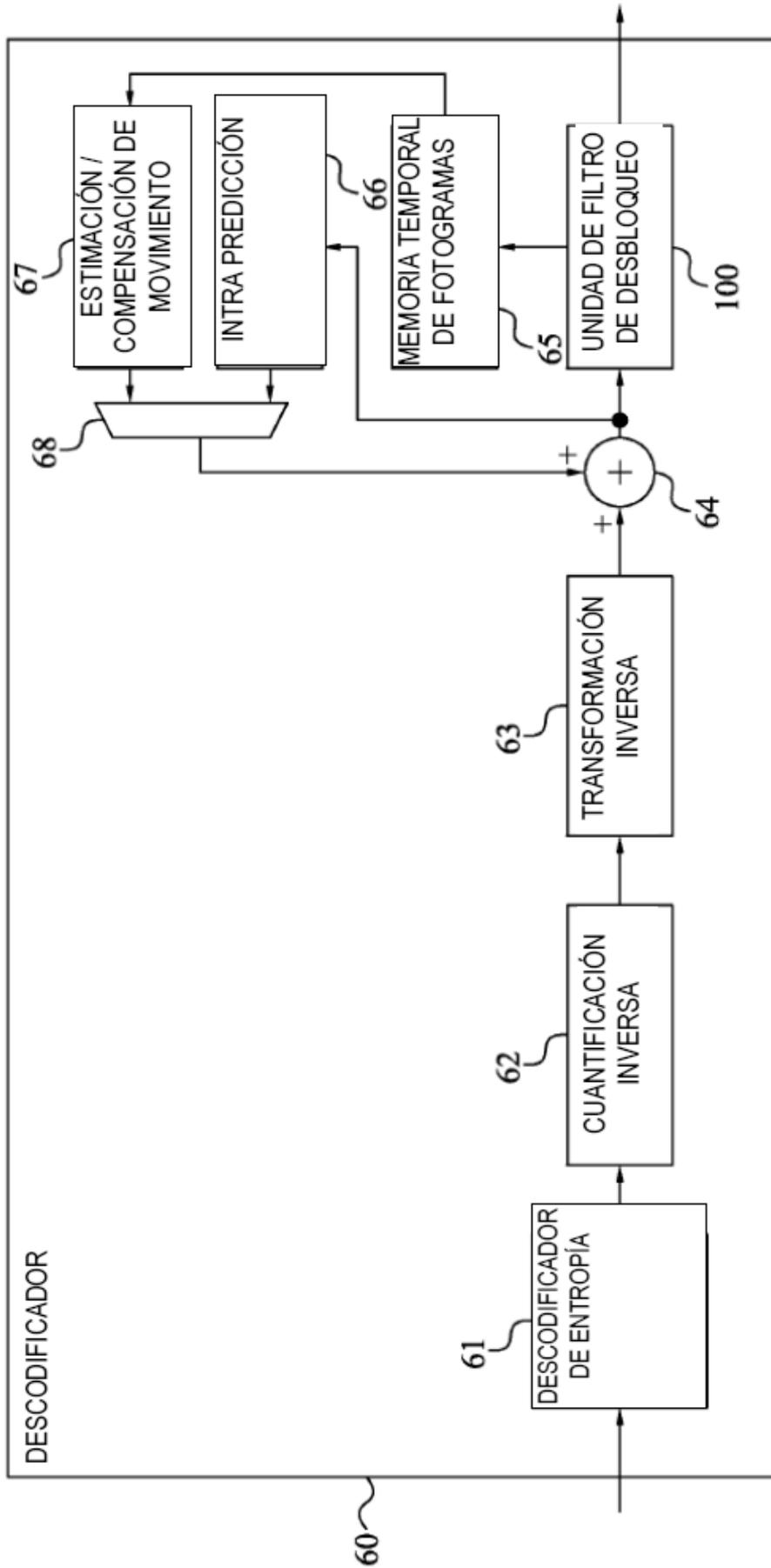


Fig. 12

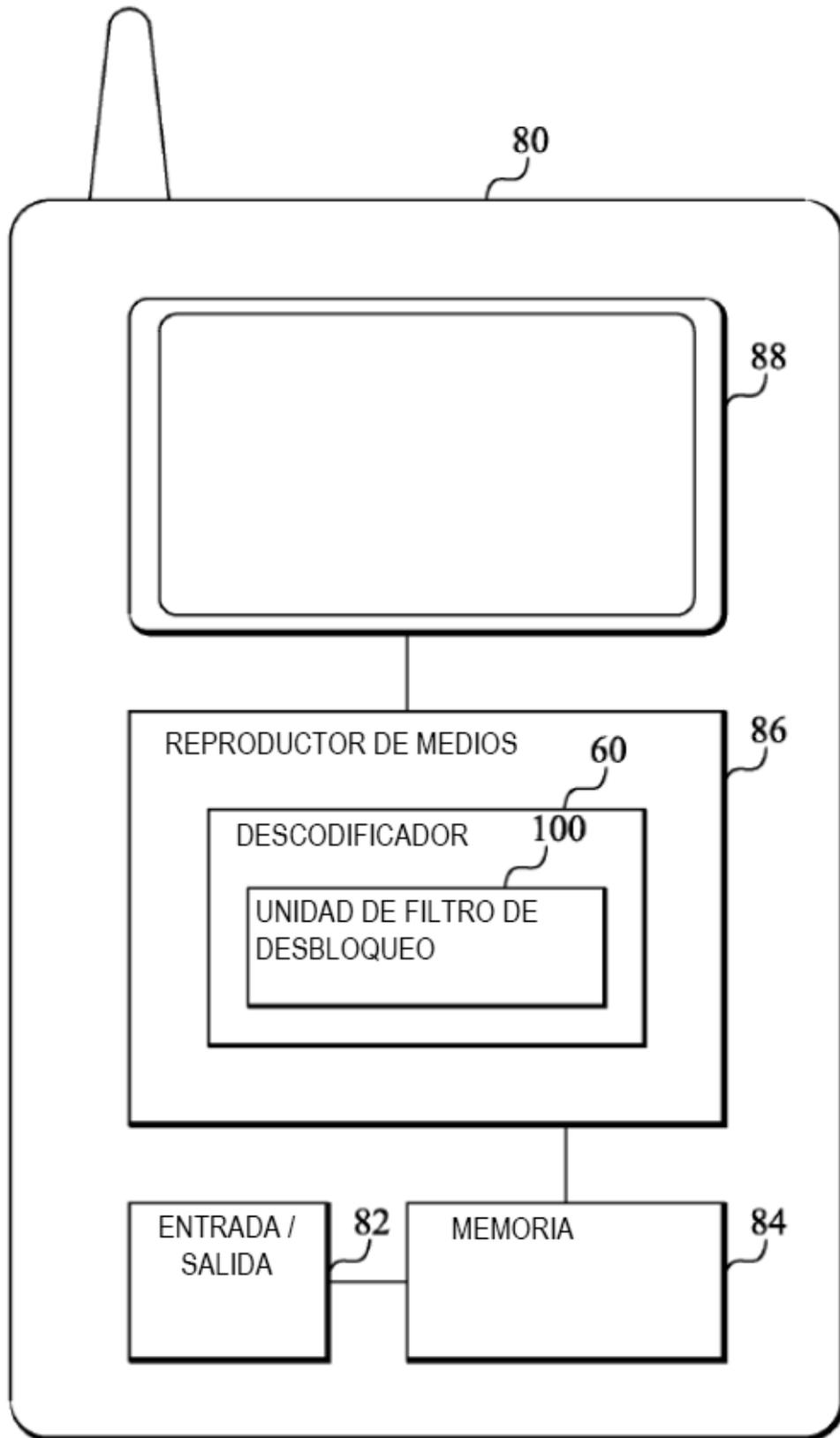


Fig. 13