



- (51) International Patent Classification:
H04N 7/26 (2006.01) *H04N 7/50* (2006.01)
- (21) International Application Number:
PCT/SE2013/050833
- (22) International Filing Date:
30 June 2013 (30.06.2013)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
61/667,505 3 July 2012 (03.07.2012) US
- (71) Applicant: TELEFONAKTIEBOLAGET L M ERICSSON (PUBL) [SE/SE]; SE-164 83 Stockholm (SE).
- (72) Inventor: NORKIN, Andrey; Blomgatan 6, SE-16960 Solna (SE).
- (74) Agent: EGRELIUS, Fredrik; Ericsson AB, Patent Unit Kista DSM, SE-16480 Stockholm (SE).
- (81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM,

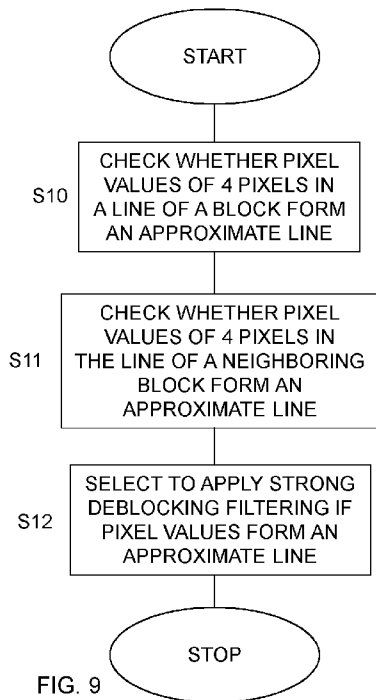
AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

- (84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:
— with international search report (Art. 21(3))

[Continued on next page]

(54) Title: STRONG DEBLOCKING FILTERING DECISIONS



(57) Abstract: A deblocking filtering control comprises checking whether pixel values of four pixels (12, 22) in a line (15) of pixels (12, 22) in a block (10) of pixels (12) and in a neighboring block (20) of pixels (22) form an approximate line. If the pixel values of the four pixels (12, 22) in the line (15) of pixels (12, 20) in both the block (10) of pixels (12) and in the neighboring block (20) of pixels (22) form an approximate line the deblocking filtering control selects to apply strong deblocking filtering to pixel values in the line (15) of pixels (12, 22). The subject and objective quality of the deblocking filtering is thereby improved by applying strong deblocking filter to areas where the signal in a picture (1) not only has the form of a flat line but also form a ramp.

WO 2014/007736 A1

- *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*

STRONG DEBLOCKING FILTERING DECISIONS

TECHNICAL FIELD

The present embodiments generally relate to deblocking filtering and in particular to controlling
5 deblocking filtering over a block boundary between neighboring blocks of pixels in a picture.

BACKGROUND

Deblocking filters are used in video coding standards in order to combat blocking artifacts. The
blocking artifacts arise because the original video is split into blocks which are processed relatively
10 independently. The blocking artifacts can arise due to different intra prediction of blocks, quantization
effects and motion compensation. Two particular variants of deblocking are described below.

These descriptions assume a vertical block boundary or border. The process is also done in the same
way for horizontal block boundaries.

15

H.264 deblocking

In state of the art video coding such as H.264 there is an adaptive de-blocking filter/loop filter after
prediction and residual reconstruction, but before storage of the reconstruction for later reference when
encoding or decoding subsequent frames. The deblocking filtering consists of several steps such as
20 filter decisions, filtering operations, a clipping function and changes of pixel values. The decision to
filter the border or not is made based on evaluating several conditions. Filter decisions depend on
macro block (MB) type, motion vector (MV) difference between neighboring blocks, whether
neighboring blocks have coded residuals and on the local structure of the current and/or neighboring
blocks.

25

The amount of filtering for a pixel depends on the position of that pixel relative to the block boundary
and on the quantization parameter (QP) value used for residual coding. Here below a to h represent
pixel values across a vertical block boundary.

30

a b c d | e f g h

The filter decision is based on comparing three pixel differences with three thresholds. The thresholds
are adapted to the QP.

If the following conditions are fulfilled the filtering is done:

$$\begin{aligned} & \text{abs}(d-e) < \text{thr1}, \\ & \text{abs}(c-d) < \text{thr2}, \text{ and} \\ 5 \quad & \text{abs}(e-f) < \text{thr2} \end{aligned}$$

where thr1 and thr2 are functions of QP.

There are two filtering modes in H.264. In the first filtering mode (normal filtering), filtering can be
10 described with a delta value that the filtering changes the current pixel value with. The filtering for the
pixel closest to the block boundary is:

$$d' = d + \text{delta} \text{ and } e' = e - \text{delta}$$

15 where delta has been clipped off to a threshold $\pm \text{thr3}$ to a value that is constrained by the QP. d' is here
the pixel value at position d after filtering and e' is the pixel value after filtering at position e. More
filtering is allowed for high QP than for low QP.

Clipping can be described as:

$$20 \quad \text{delta_clipped} = \max(-\text{thr3}, \min(\text{thr3}, \text{delta}))$$

where thr3 is controlling the filter strength. A larger value of thr3 means that the filtering is stronger,
which in turns means that a stronger low-pass filtering effect will happen.

25

The filter strength can be increased if any of the following two conditions also holds:

$$\text{abs}(b-d) < \text{thr2} \text{ and } \text{abs}(e-g) < \text{thr2}$$

30 The filter strength is adapted by clipping the delta less, e.g. to allow for more variation.

The second filtering mode (strong filtering) is applied for intra macroblock boundaries only, when the
following condition is fulfilled:

$\text{abs}(d-e) < \text{thr1}/4$.

The thresholds thr1, thr2 and thr3 are derived from table lookup using QP as index. Each slice can contain modifications of thr2 and thr3 using *slice_beta_offset_div2* and thr1 using *slice_alpha_c0_offset_div2*. The slice parameters $2 \times \text{slice_beta_offset_div2}$ and $2 \times \text{slice_alpha_c0_offset_div2}$ are added to the current QP index before table lookup of thr2/thr3 and thr1 respectively.

Deblocking in HEVC draft

10 Here below p_0 to p_3 and q_0 to q_3 represent pixel values across a vertical block boundary.

$p_3 \ p_2 \ p_1 \ p_0 \ | \ q_0 \ q_1 \ q_2 \ q_3$

In the draft HEVC specification, the deblocking filter works differently than H.264. The filtering is performed if at least one of the blocks on the side of the border is intra, or has non-zero coefficients, or the difference between the motion vector components of the blocks is greater than or equal to one integer pixel. For example, if one is filtering the border between the blocks A and B below, then the following condition should satisfy for the block boundary to be filtered:

20

A				B			
p_{3_0}	p_{2_0}	p_{1_0}	p_{0_0}	q_{0_0}	q_{1_0}	q_{2_0}	q_{3_0}
p_{3_1}	p_{2_1}	p_{1_1}	p_{0_1}	q_{0_1}	q_{1_1}	q_{2_1}	q_{3_1}
p_{3_2}	p_{2_2}	p_{1_2}	p_{0_2}	q_{0_2}	q_{1_2}	q_{2_2}	q_{3_2}
p_{3_3}	p_{2_3}	p_{1_3}	p_{0_3}	q_{0_3}	q_{1_3}	q_{2_3}	q_{3_3}

$$dp0 = | p_{2_0} - 2 \times p_{1_0} + p_{0_0} | \tag{1}$$

$$dp3 = | p_{2_3} - 2 \times p_{1_3} + p_{0_3} |$$

25 $dq0 = | q_{2_0} - 2 \times q_{1_0} + q_{0_0} |$

$$dq3 = | q_{2_3} - 2 \times q_{1_3} + q_{0_3} |$$

$$dpq0 = dp0 + dq0$$

$$dpq3 = dp3 + dq3$$

$$dp = dp0 + dp3$$

30 $dq = dq0 + dq3$

The variable d is derived as follows:

$$d = dpq0 + dpq3 \tag{2}$$

5

The deblocking filtering is performed on the block boundary for lines $i = 0..3$ if the following condition holds:

$$d < \beta \tag{3}$$

10

where β depends on QP. In the draft HEVC specification, there is a table, see Table 1 below, for looking up the value of β using QP as the table index. β increases with increasing QP.

Table 1 – Derivation of threshold variables β and t_c from input QP

QP	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
β	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	7	8
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
β	9	10	11	12	13	14	15	16	17	18	20	22	24	26	28	30	32	34	36
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	
β	38	40	42	44	46	48	50	52	54	56	58	60	62	64	64	64	64	64	
t_c	5	5	6	6	7	8	9	9	10	10	11	11	12	12	13	13	14	14	

15

If the condition in equation (3) above is fulfilled and filtering is done between blocks A and B, one of two types of filtering (weak or strong filtering) is performed. The choice between the strong and the weak filtering is done separately for each line depending on the following conditions. For lines $i = 0,3$, strong filtering is performed if all the following conditions are true, otherwise, weak filtering is performed:

20

$$2 \times dpq_i < (\beta >> 2) \tag{4}$$

$$\text{and } (|p3_i - p0_i| + |q0_i - q3_i|) < (\beta >> 3) \tag{5}$$

$$\text{and } |p0_i - q0_i| < ((5 * t_c + 1) >> 1), \tag{6}$$

where t_c and β depend on QP as shown in Table 1. The t_c is calculated as $t_c \times (QP+2)$ when A or B has $PredMode == MODE_INTRA$.

The two filtering modes (weak and strong filtering) in the HEVC draft look like in the following:

5

Weak filtering

Weak filtering is performed based on the above conditions. The actual filtering works by computing an offset (Δ) for each of the lines i that the weak filter has been selected for. The following weak filtering procedure is applied for every line, where it has been chosen. In the following algorithm, the variables
 10 $p0..p2$ and $q0..q2$ are assigned the following values (from row/column i) where $p0=p0_i$, $p1=p1_i$,
 $p2=p2_i$, $q0=q0_i$, $q1=q1_i$, $q2=q2_i$. One can see that if the weak filtering is performed, one to two pixels
 are modified at each side of the block boundary:

$$\Delta = (9 \times (q0 - p0) - 3 \times (q1 - p1) + 8) \gg 4$$

15 if(abs(Δ) < $10 \times t_c$)

{

$$\Delta = \text{Clip3}(-t_c, t_c, \Delta)$$

$$p0' = \text{Clip1}_Y(p0 + \Delta)$$

$$q0' = \text{Clip1}_Y(q0 - \Delta)$$

20 if($dp < (\beta + (\beta \gg 1)) \gg 3$)

(7)

{

$$\Delta p = \text{Clip3}(-(\beta \gg 1), \beta \gg 1, (((p2 + p0 + 1) \gg 1) - p1 + \Delta) \gg 1)$$

$$p1' = \text{Clip1}_Y(p1 + \Delta p)$$

}

25 if($dq < (\beta + (\beta \gg 1)) \gg 3$)

(8)

{

$$\Delta q = \text{Clip3}(-(\beta \gg 1), \beta \gg 1, (((q2 + q0 + 1) \gg 1) - q1 - \Delta) \gg 1)$$

$$q1' = \text{Clip1}_Y(q1 + \Delta q)$$

}

30 }

where Clip is defined as $x' = \text{Clip3}(A, B, x)$: $x' = x$ or if $x < A$ $x' = A$ or if $x > B$ $x' = B$ and $\text{Clip1}_Y(x) = \text{Clip3}(0, (1 \ll \text{BitDepth}_Y) - 1, x)$, where BitDepth_Y is the bit depth, for example 8 or 10.

Strong filtering

Strong filtering mode is performed for a line i of pixels by the following set of operations, where $p_0=p_{0i}$, $p_1=p_{1i}$, $p_2=p_{2i}$, $q_0=q_{0i}$, $q_1=q_{1i}$, $q_2=q_{2i}$:

$$\begin{aligned}
 5 \quad p_0' &= \text{Clip3}(p_0-2 \times tc, p_0+2 \times tc, (p_2 + 2 \times p_1 + 2 \times p_0 + 2 \times q_0 + q_1 + 4) \gg 3) \\
 p_1' &= \text{Clip3}(p_1-2 \times tc, p_1+2 \times tc, (p_2 + p_1 + p_0 + q_0 + 2) \gg 2) \\
 p_2' &= \text{Clip3}(p_2-2 \times tc, p_2+2 \times tc, (2 \times p_3 + 3 \times p_2 + p_1 + p_0 + q_0 + 4) \gg 3) \\
 q_0' &= \text{Clip3}(q_0-2 \times tc, q_0+2 \times tc, (p_1 + 2 \times p_0 + 2 \times q_0 + 2 \times q_1 + q_2 + 4) \gg 3) \\
 q_1' &= \text{Clip3}(q_1-2 \times tc, q_1+2 \times tc, (p_0 + q_0 + q_1 + q_2 + 2) \gg 2) \\
 10 \quad q_2' &= \text{Clip3}(q_2-2 \times tc, q_2+2 \times tc, (p_0 + q_0 + q_1 + 3 \times q_2 + 2 \times q_3 + 4) \gg 3)
 \end{aligned}$$

The decision for applying a strong deblocking filter in HEVC based on equations (4)-(6) makes sure that the strong deblocking filtering is only applied when the signal on both sides of the block boundary is flat. However, the prior art filter decision of whether to apply strong or weak deblocking filtering has
 15 disadvantages in terms of selecting weak deblocking filtering for some blocks and block boundaries even when strong deblocking filtering might be more appropriate. Hence, the prior art filter decision can result in insufficient deblocking filtering, which causes a reduction in subjective and/or objective quality of the picture and video.

20

SUMMARY

It is a general object to provide an improved deblocking filtering control.

It is a particular object to provide an improved strong filtering decision.

25 These and other objects are met by embodiment as disclosed herein.

An aspect of the embodiment relates to a deblocking filtering control method. The method comprises checking whether pixel values of four pixels in a line of pixel in a block of pixels form an approximate line. The method also comprises checking whether pixel values of four pixels in the line of pixels in a
 30 neighboring block of pixels form an approximate line. The method further comprises selecting to apply strong deblocking filtering to pixel values in the line of pixels in the block of pixels and the neighboring block of pixels if the pixel values of the four pixels in the block of pixels form an approximate line and if the pixel values of the four pixels in the neighboring block of pixels form an approximate line.

Another aspect of the embodiments relates to a filtering control device comprising a determining unit configured to determine whether pixel values of four pixels in a line of pixel in a block of pixels form an approximate line. The determining unit is also configured to determine whether pixel values of four pixels in the line of pixels in a neighboring block of pixels form an approximate line. The filtering control
5 device also comprises a processing unit configured to select to apply strong deblocking filtering to pixel values in the line of pixels in the block of pixels and the neighboring block of pixels if the pixel values of the four pixels in the block of pixels form an approximate line and if the pixel values of the four pixels in the neighboring block of pixels form an approximate line.

10 Further aspects of the embodiments relate to an encoder comprising a filtering control device according to above, a decoder comprising a filtering control device according to above, a user equipment comprising an encoder and/or a decoder according to above and a network device being or belonging to a network node in a communication network. The network device comprises an encoder and/or a decoder according to above.

15

Yet another aspect of the embodiments relates to a computer comprising a processing unit and a computer program product comprising a computer program. The computer program comprises code means, which when run on the processing unit, causes the processing unit to check whether pixel values of four pixels in a line of pixels in a block of pixels form an approximate line. The computer
20 program also causes the processing unit to check whether pixel values of four pixels in the line of pixels in a neighboring block of pixels form an approximate line. The computer program further causes the processing unit to select to apply strong deblocking filtering to pixel values in the line of pixels in the block of pixels and the neighboring block of pixels if the pixel values of the four pixels in the block of pixels form an approximate line and if the pixel values of the four pixels in the neighboring block of
25 pixels form an approximate line.

A further aspect of the embodiments relates to a computer program for filtering control. The computer program comprises code means, which when executed by a processing unit, causes the processing unit to check whether pixel values of four pixels in a line of pixels in a block of pixels form an
30 approximate line. The computer program also causes the processing unit to check whether pixel values of four pixels in the line of pixels in a neighboring block of pixels form an approximate line. The computer program further causes the processing unit to select to apply strong deblocking filtering to pixel values in the line of pixels in the block of pixels and the neighboring block of pixels if the pixel

values of the four pixels in the block of pixels form an approximate line and if the pixel values of the four pixels in the neighboring block of pixels form an approximate line.

A related aspect of the embodiments relates to a computer program product comprising computer
5 readable code means and a computer program according to above stored on the computer readable code means.

The proposed embodiments improve the subjective and/or objective quality of strong deblocking filtering, such as HEVC strong filtering. The embodiments enable application of strong deblocking filter
10 to areas where the signal in a picture has the form of a gradient or ramp. This enables application of the strong deblocking filtering to the areas where blocking artifacts might otherwise be visible.

BRIEF DESCRIPTION OF THE DRAWINGS

The embodiments, together with further objects and advantages thereof, may best be understood by
15 making reference to the following description taken together with the accompanying drawings, in which:

Figs. 1A-1C illustrate embodiments of methods performed in a filtering control (ctrl) device, a transmitter and a receiver, respectively;

20 Fig. 2 is a schematic block diagram of an encoder according to an embodiment;

Fig. 3 is a schematic block diagram of a decoder according to an embodiment;

Fig. 4 is a schematic block diagram of a user equipment according to an embodiment;

25

Fig. 5 is a schematic block diagram of a user equipment according to another embodiment;

Fig. 6 is a schematic block diagram of a network node in a communication network according to an embodiment;

30

Fig. 7 is a schematic block diagram of a filtering control (ctrl) device according to an embodiment;

Fig. 8 is a schematic block diagram of a computer according to an embodiment;

Fig. 9 is a flow diagram of a filtering control method according to an embodiment;

Fig. 10 is a flow diagram of an implementation embodiment of the selecting step in Fig. 9;

5 Fig. 11 is a flow diagram of an optional step of the filtering control method in Fig. 9;

Fig. 12 is a flow diagram of an implementation embodiment of the checking steps in Fig. 9;

Fig. 13 is a flow diagram of an implementation embodiment of the filtering control method in Fig. 9;

10

Fig. 14 is a flow diagram of an optional step and an implementation embodiment of the selecting step in Fig. 13;

Fig. 15 is a schematic block diagram of an implementation embodiment of the determining unit in Fig. 7;

15

Figs. 16A-16D illustrate examples of variations in pixel values across a block boundary between a block of pixels and a neighboring block of pixels; and

Fig. 17 schematically illustrates a picture of a video sequence with a block of pixels and a neighboring
20 block of pixels separated by a block boundary.

DETAILED DESCRIPTION

Throughout the drawings, the same reference numbers are used for similar or corresponding elements.

25 The present embodiments generally relate to deblocking filtering and in particular to controlling deblocking filtering over a block boundary or block border between neighboring blocks of pixels in a picture.

Deblocking filters are used to combat blocking artifacts that arise because the pictures in the original
30 video are split into blocks of pixels, which are processed relatively independently. Deblocking filtering is often divided into two types or modes of filtering, generally denoted weak filtering and strong filtering. A strong filter applies, which its name indicate, a stronger filtering of pixel values as compared to a weak filter.

The deblocking filtering decision proposed for HEVC and disclosed in the background section herein makes sure that the strong deblocking filtering is only applied when the signal on both sides of a block boundary is flat. This corresponds to the situation illustrated in Fig. 16A, where the bold line represents the block boundary and the horizontal lines represent the pixel values in a row of pixels in a block of 5 pixels to the left of the block boundary and the pixel values in the row of pixels in a neighboring block of pixels to the right of the block boundary.

However, this means that strong filtering is not applied according to the prior art if the signal on one or both sides of the block boundary has the form of a ramp even if the ramp is smooth. Figs. 16B and 16C illustrate such situations where the pixel values in a row of pixels form a ramp in both the block of pixels and in the neighboring block of pixels.

This means that the strong deblocking filtering of HEVC is not applied to the smooth areas which have the form of a ramp, e.g. with a gradual change in pixel intensity, as shown in Figs. 16B and 16C. This in turn results in insufficient deblocking filtering, which causes a reduction in subjective and/or objective quality of the picture and video.

An improvement to the strong deblocking filter decision, in particular for HEVC, is proposed by the embodiments. In particular, the embodiments use the criterion that the signal on both sides of the block boundary is smooth. However, there are no requirements that the signal is flat. Hence, an embodiment involves, as shown in Fig. 1A illustrating a deblocking filter control method performed in a filtering control (ctrl) device. The method comprises checking, in step S1, if the signal on both sides of the block boundary is smooth. In particular, a strong deblocking filtering decision checks or determines whether four pixels on both sides of the block boundary form an approximate line.

25

Fig. 9 illustrates a flow diagram of a deblocking filter control method according to an embodiment. The method starts in step S10 which comprises checking or determining whether pixel values of four pixels in a line of pixels in a block of pixels form an approximate line. Step S11 comprises checking or determining whether pixel values of four pixels in the line of pixels in a neighboring block of pixels form an approximate line. The following step S12 comprises selecting to apply strong deblocking filtering to pixel values in the line of pixels in the block of pixels and the neighboring block of pixels if the pixel values of the four pixels in the block of pixels form an approximate line and if the pixel values of the four pixels in the neighboring block of pixels form an approximate line.

30

Steps S10 and S11 of Fig. 9 can be performed serially in any order, i.e. S10 prior to step S11 or step S11 prior to step S10. The two steps S10 and S11 can alternatively be performed at least partly in parallel.

5 In the art, filtering pixel values of pixels in a line of pixels in a block of pixels is sometimes denoted as filtering sample values of samples in a line of samples in a block of samples. Herein, pixel values and pixels are used throughout the description but also encompass alternative notations used in the art, such as sample values and samples. A pixel value (sample value) generally represents a color value of the pixel (sample). Different color formats are available in the art including luminance + chrominance,
10 red + green + blue (RGB) etc. Video coding generally use the color format with one luminance component and two chrominance components for each pixel. Traditionally, luminance component filtering and chrominance component filtering are done separately possibly employing different filtering decisions and different de-blocking filters. The embodiments can be applied to filtering the luminance component, filtering a chrominance component or filtering both the luminance component and a
15 chrominance component. In a particular embodiment, the embodiments are applied to achieve luminance or luma filtering. Hence, in a particular embodiment a pixel value (sample value) represents a luminance value.

Fig. 17 schematically illustrates a picture 1, such a picture 1 of a video sequence, in which the pixels
20 12, 22 have been divided into blocks 10, 20 of pixels 12, 22. The figure also shows a block boundary or border 2 between the block 10 of pixels 12 and its neighboring block 20 of pixels 22. Hence, the block 10 of pixels 12 and the neighboring block 20 of pixels 22 are two adjacent blocks 10, 20 of pixels 12, 22 in the picture 1 and are separate by the block boundary 2. Fig. 17 illustrates the case where the two blocks 10, 20 of pixels 12, 22 are positioned next to each other in the picture 1 and the block boundary
25 2 therefore is a vertical block boundary. In an alternative approach, the two blocks 10, 20 of pixels 12, 22 are positioned in the picture 1 with one of the blocks 10, 20 of pixels 12, 22 positioned above the other block of pixels. In this alternative approach the block boundary 2 is a horizontal block boundary.

Step S10 of Fig. 9 checks whether pixel values of four pixels 12 in a line 15 of pixels 12, 22 in the block
30 10 of pixels 12 form an approximate line. This line 15 of pixels 12, 22 is either a row of pixels 12, 22 as shown in Fig. 17 for a vertical block boundary 2 or a column of pixels 12, 22 for a horizontal block boundary. The row or column of pixels 12, 22 extend over the vertical or horizontal block boundary 2 to pass over and encompass both pixels 12 in the block 10 of pixels 12 and pixels 22 in the neighboring block 20 of pixels 22.

Step S11 correspondingly checks whether pixel values of four pixels 22 in the line 15 of pixels 12, 22 in the neighboring block 20 of pixels 22 form an approximate line. The pixel values checked or investigated in steps S10 and S11 thereby belong to the same line 15 of pixels 12, 22 but belong to 5 different blocks 10, 20 of pixels 12, 22 in the picture 1.

In a particular approach, the pixel values of the four pixels 12, 22 checked in steps S10 and S11 are the pixel values of the four pixels 12, 22 closest to the block boundary 2 in the line 15 of pixels 12, 22 in the two blocks 10, 20 of pixels 12, 22. Generally, the pixel value of pixel number j relative to, i.e. 10 counting from, the block boundary 2 in line number i in the block 10 of pixels 12 is denoted p_j ; herein. The corresponding pixel value of pixel number j relative to the block boundary 2 in line number i in the neighboring block 20 of pixels 22 is denoted q_j . In such a case, step S10 preferably comprises checking whether pixel values p_{0i} , p_{1i} , p_{2i} and p_{3i} in line number i form an approximate line and step S11 preferably comprises checking whether pixel values q_{0i} , q_{1i} , q_{2i} and q_{3i} in line number i form an 15 approximate line for at least one selected value of i . In an embodiment, in particular applicable to HEVC, $i=0$ or 3 or $i=0\dots3$.

If the pixel values of the four pixels 12, 22 in the line 15 of pixels 12, 22 in both the block 10 of pixels 12 and the neighboring block 20 of pixels 22 form an approximate line as checked in steps S10 and S11 20 step S12 of Fig. 9 comprises selecting to apply strong deblocking filtering to pixel values in the line of pixels 12, 22 in both blocks 10, 20 of pixels 12, 22.

The deblocking filter control method of Fig. 9 thereby selects to apply strong deblocking filtering for all of the situations shown in Figs. 16A to 16C, whereas the prior art filter decisions of HEVC will only 25 select strong deblocking filtering for the situation in Fig. 16A but not in Figs. 16B and 16C.

Hence, the embodiments not only select to apply strong deblocking filtering if the pixels values in the line 15 of pixels 12, 22 for an approximate flat or horizontal line, as in Fig. 16A, but also select to apply strong deblocking filtering if pixel values in the line 15 of pixels 12, 22 form a ramp, as in Figs. 16B and 30 16C. In Fig. 16B the pixel values form an approximately continuous ramp when traveling along the line 15 of pixels 12, 22 over the block boundary 2. Fig. 16C illustrates an alternative approach in which the four pixel values in the line 15 of pixels 12, 22 in the block 10 of pixels 12 form a ramp and the four pixel values in the line 15 of pixels 12, 22 in the neighboring block 20 of pixels 22 form a ramp, which may have a different slope as compared to the ramp in the block 10 of pixels 12. The embodiments

also select to apply strong deblocking filtering if the four pixel values in the line 15 of pixels 12, 22 form an approximate flat or horizontal line in one of the two blocks 10, 20 of pixels 12, 22 and the four pixel values in the line 15 of pixels 12, 22 form an approximate ramp in the other of the two blocks 10, 20 of pixels 12, 22.

5

In a particular embodiment, step S10 of Fig. 9 thereby comprises checking or determining whether the pixel values of the four pixels 12 in the line 15 of pixels 12, 22 in the block 10 of pixels 12 form a flat line or ramp, or at least form an approximate flat line or ramp. Correspondingly, step S11 then comprises checking or determining whether the pixel values of the four pixels 22 in the line 15 of pixels 12, 22 in
10 the neighboring block 20 of pixels 22 form a flat line or ramp, or at least form an approximate flat line or ramp. Step S12 preferably comprises, in this particular embodiment, selecting to apply strong deblocking filtering if the pixel values of the four pixels 12 in the block 10 of pixels 12 form a flat line or a ramp, or at least an approximate flat line or ramp, and if the pixel values of the four pixels 22 in the neighboring block 20 of pixels 22 form a flat line or a ramp, or at least an approximate flat line or ramp.

15

Fig. 10 is a flow diagram illustrating an optional step of the deblocking filter control method of Fig. 9 if the condition in step S12 is not met. The method then continues from step S11 in Fig. 9. Step S20 comprises selecting to apply weak deblocking filtering to pixel values in the line 15 of pixels 12, 22 in the block 10 of pixels 12 and the neighboring block 20 of pixels 22 if the pixel values of the four pixels
20 12 in the block 10 of pixels 12 do not form an approximate line and/or if the pixel values of the four pixels 22 in the neighboring block 20 of pixels 22 do not form an approximate line.

Hence, if the four pixel values on at least one of the sides of the block boundary 2 do not form an approximate line, such as a flat line or a ramp, then strong deblocking filtering is not appropriate for the
25 pixel values in the current line 15 of pixels 12, 22. In clear contrast, weak deblocking filtering is instead selected in step S20. The reason being that it is likely that there will be a structure in the line 2 of pixels 12, 22, such as an edge between two different objects or areas in the picture 1 or a highly detailed texture. This edge or texture should thereby not be filtered away during the deblocking filtering. Accordingly, a weak deblocking filter is more appropriate as compared to using a strong deblocking
30 filter.

Weak filtering is preferably performed as disclosed in the section weak filtering for HEVC in the background section and strong filtering may be performed as disclosed in the background section under section strong filtering for HEVC.

In an embodiment, at least one additional criterion is used to discriminate between selecting to apply strong deblocking filtering and weak deblocking filtering. An additional check to make sure that there is no large step between the pixel values of the block boundary 2 can thereby also be used. This check
5 makes sure that the difference between the pixel values on both sides of the block boundary 2 is not due to a natural edge in the original signal.

This situation is illustrated in Fig. 16D. In this figure the four pixel values in the line 15 of pixels 12, 22 form a ramp and thereby an approximate line on either side of the block boundary 2. However, there is
10 a significant difference in pixel value of the pixel 12 closest to the block boundary 2 in the block 10 of pixels 12 and the pixel value of the pixel 22 closest to the block boundary 2 in the neighboring block 20 of pixels 22. Hence, when traveling along the line 15 of pixels 12, 22 there is a significant step or jump in pixel values at the block boundary 2. This step may be due to a natural edge between different objects or areas in the picture.

15

A particular embodiment of the deblocking filter control method thereby comprises an additional step S30 as shown in Fig. 11 to identify a situation similar to the one shown in Fig. 16D. The method continues from step S11 of Fig. 9 but could alternatively continue from step S10 or indeed the method could start with step S30. This step S30 comprises checking or determining whether a difference
20 between i) a pixel value of a first pixel 12, relative to the block boundary 2 between the block 10 of pixels 12 and the neighboring block 20 of pixels 22, in the line 15 of pixels 12, 22 in the block of pixels 12 and ii) a pixel value of a first pixel 22, relative to the block boundary 2, in the line 15 of pixels 12, 22 in the neighboring block 20 of pixels 22 is smaller than a threshold value.

25 The method then continues back to Fig. 9 where step S12 comprises selecting to apply strong deblocking filtering if the pixel values of the four pixels 12 in the block 10 of pixels 12 form an approximate line, if the pixel values of the four pixels 22 in the neighboring block 20 of pixels 22 form an approximate line and if the difference checked in step S30 is smaller than the threshold value.

30 This means that step S30 checks the difference $p_{0i} - q_{0i}$ and in particular checks whether such a difference is smaller than a threshold value.

This embodiment combining Fig. 9 and Fig. 11 is able to differentiate a situation with a larger step at the block boundary 2 due to a natural edge, see Fig. 16D, from a situation where there is a smaller step at the block boundary 2, see Figs. 16A to 16C, likely because of a block artifact.

- 5 It is proposed, in an embodiment, to use a combination of the strong filtering decision from (4) in the background section, which checks that the first three pixels 12, 22 from the block boundary 2 form a straight line, e.g. a flat line or a ramp, or some similar decision that checks that the first three pixels 12, 22 from the block boundary 2 form a line. Another criterion checks that the values of the second, the third and the fourth pixels also form a straight line. For example, checking that pixels p_{1i} , p_{2i} and p_{3i}
- 10 approximately form a straight line, e.g. a ramp, in addition to check that the pixels p_{0i} , p_{1i} and p_{2i} form a straight line, e.g. a ramp, makes sure that all four pixels p_{0i} , p_{1i} , p_{2i} and p_{3i} form an approximately straight line. If the same criterion also holds for pixels q_{0i} , q_{1i} , q_{2i} and q_{3i} then the strong deblocking filter can be applied. In addition, a criterion similar to the one from (6) in the background section can be applied to make sure that the step at the block boundary 2 is not a natural edge, i.e. that the absolute
- 15 difference between the pixel values is below a certain threshold.

Fig. 12 is a flow diagram illustrating an example of steps S10 and S11 according to this embodiment. The method starts in step S40, which comprises checking or determining whether pixel values of a first pixel 12, a second pixel and a third pixel, relative to the block boundary 2 between the block 10 of pixels 12 and the neighboring block 20 of pixels 22, in the line 15 of pixels 12, 22 in the block 10 of pixels 12 form an approximate line. Step S41 comprises checking or determining whether pixel values of the second pixel, the third pixel and a fourth pixel, relative to the block boundary 2, in the line 15 of pixels 12, 22 in the block 10 of pixels 12 form an approximate line. Step S41 comprises checking or determining whether pixel values of a first pixel 22, a second pixel and a third pixel, relative to the block

25 boundary 2, in the line 15 of pixels 12, 22 in the neighboring block 20 of pixels 22 form an approximate line. Finally, step S42 comprises checking or determining whether pixel values of the second pixel, the third pixel and a fourth pixel, relative to the block boundary 2, in the line 15 of pixels 12, 22 in the neighboring block 20 of pixels 22 form an approximate line.

- 30 Steps S40 to S43 can be performed serially in any order or at least partly in parallel.

The method then continues to step S12 of Fig. 9. In this embodiment step S12 comprises selecting to apply strong deblocking filtering if the pixel values of the first pixel 12, the second pixel and the third pixel in the block 10 of pixels form an approximate line, if the pixel values of the second pixel, the third

pixel and the fourth pixel in the block 10 of pixels 12 form an approximate line, if the pixel values of the first pixel 22, the second pixel and the third pixel in the neighboring block 20 of pixels 22 form an approximate line and if the pixel values of the second pixel, the third pixel and the fourth pixel in the neighboring block 20 of pixels 22 form an approximate line.

5

Fig. 13 is a flow diagram illustrating a particular implementation example of the deblocking filter control method. The method starts in step S50, which comprises calculating $dp_i = |p_{2i} - 2 \times p_{1i} + p_{0i}|$. A next step S51 comprises calculating $dp_{i_side} = |p_{3i} - 2 \times p_{2i} + p_{1i}|$. The following step S52 comprises calculating $dq_i = |q_{2i} - 2 \times q_{1i} + q_{0i}|$ and step S53 comprises calculating $dq_{i_side} = |q_{3i} - 2 \times q_{2i} + q_{1i}|$.

10

Steps S50 to S53 can be performed serially in any or order or at least partly in parallel.

As previously disclosed herein wherein p_{0i} denotes a pixel value of the first pixel 12 in the line 15 of pixels 12, 22 in the block 10 of pixels 12, p_{1i} denotes a pixel value of the second pixel in the line 15 of pixels 12, 22 in the block 10 of pixels 12, p_{2i} denotes a pixel value of the third pixel in the line 15 of pixels 12, 22 in the block 10 of pixels 12, p_{3i} denotes a pixel value of the fourth pixel in the line 15 of pixels 12, 22 in the block 10 of pixels 12, q_{0i} denotes a pixel value of the first pixel 22 in the line 15 of pixels 12, 22 in the neighboring block 20 of pixels 22, q_{1i} denotes a pixel value of the second pixel in the line 15 of pixels 12, 22 in the neighboring block 20 of pixels 22, q_{2i} denotes a pixel value of the third pixel in the line 15 of pixels 12, 22 in the neighboring block 20 of pixels 22 and q_{3i} denotes a pixel value of the fourth pixel in the line 15 of pixels 12, 22 in the neighboring block 20 of pixels 22.

The method then continues to step S54, which comprises selecting to apply strong deblocking filtering if $(dp_i + dq_i) < Thr1$ and $(dp_{i_side} + dq_{i_side}) < Thr2$, wherein $Thr1$ denotes a first threshold value and $Thr2$ denotes a second threshold value. Hence, if the sum of the parameter values calculated in steps S50 and S52 based on the three first pixel values in the line 15 of pixels 12, 22 on either side of the block boundary 2 is smaller than a first threshold value and if the sum of the parameter values calculated in steps S51 and S53 based on the second to fourth pixel values in the line 15 of pixels 12, 22 on either side of the block boundary 2 is smaller than a second threshold value strong deblocking filtering is selected in step S54. The method then continues to step S55, where strong deblocking filter is applied to pixel values in the line 15 of pixels 12, 22, such as disclosed in the background section. However, if $(dp_i + dq_i) \geq Thr1$ and/or $(dp_{i_side} + dq_{i_side}) \geq Thr2$ the method instead continues from step S54 to step S56. This step S56 comprises applying weak deblocking filter to pixel values in the line of pixels 12, 22, such as disclosed in the background section.

If the pixel values of the four pixels 12 in the line 15 of pixels 12, 22 in the block 10 of pixels 12 and in the neighboring block 20 of pixels 22 form a flat line then $p_0=p_1=p_2=p_3$ and $q_0=q_1=q_2=q_3$. This in turn implies that $dp_i=0$, $dq_i=0$, $dp_{i_side}=0$ and $dq_{i_side}=0$. Accordingly, the sums $(dp_i + dq_i)$ and $(dp_{i_side} + dq_{i_side})$ will also be zero. Correspondingly, if the pixel values for a ramp, for instance $p_0=4$, $p_1=3$, $p_2=2$, $p_3=1$ and $q_0=6$, $q_1=7$, $q_2=8$ and $q_3=9$ then $dp_i=0$, $dq_i=0$, $dp_{i_side}=0$ and $dq_{i_side}=0$ and also the sums $(dp_i + dq_i)$ and $(dp_{i_side} + dq_{i_side})$ will be zero.

A particular embodiment of step S54 comprises selecting to apply strong deblocking filtering if $2 \times (dp_i + dq_i) < (\beta \gg 2)$ and $2 \times (dp_{i_side} + dq_{i_side}) < (\beta \gg 2)$, wherein \gg represents a right shift operator defined as $a \gg b = \left\lfloor \frac{a}{2^b} \right\rfloor$ and $\lfloor c \rfloor$ is a largest integer not greater than c and β is a parameter that depends on a quantization parameter value associated with the block 10 of pixels 12. Hence, in this embodiment $Thr1 = (\beta \gg 2) / 2$ and $Thr2 = (\beta \gg 2) / 2$.

$2 \times (dp_i + dq_i) < (\beta \gg 2)$ is equivalent to $((dp_i + dq_i) \ll 1) < (\beta \gg 2)$, wherein \ll represents a left shift operator defined as $a \ll b = a \times 2^b$. Correspondingly, $2 \times (dp_{i_side} + dq_{i_side}) < (\beta \gg 2)$ is equivalent to $((dp_{i_side} + dq_{i_side}) \ll 1) < (\beta \gg 2)$.

The embodiment discussed above and disclosed in Fig. 13 can optionally be combined with an additional criterion to identify a step in pixel values at the block boundary 2. Such an approach is shown in Fig. 14. The method then continues from step S54 in Fig. 13. A next step S60 comprises calculating $| p_0 - q_0 |$.

This step S60 can be performed serially relative to or at least partly in parallel with any of the steps S50 to S53.

The method then continues to step S61, which comprises selecting to apply strong deblocking filtering if $(dp_i + dq_i) < Thr1$, $(dp_{i_side} + dq_{i_side}) < Thr2$ and $| p_0 - q_0 | < Thr3$, wherein $Thr3$ denotes a third threshold value.

30

In a particular embodiment $Thr3$ is $((5 \times t_c + 1) \gg 1)$, wherein t_c is a parameter that depends on the quantization parameter value associated with the block 10 of pixels 12. Step S61 preferably comprises,

in this particular embodiment, selecting to apply strong deblocking filtering if $2 \times (dp_i + dq_i) < (\beta \gg 2)$, $2 \times (dp_{i_side} + dq_{i_side}) < (\beta \gg 2)$ and $|p_{0i} - q_{0i}| < ((5 \times t_c + 1) \gg 1)$.

In another particular embodiment Th_3 is $(t_c \ll 4)$. Step S61 preferably comprises, in this particular embodiment, selecting to apply strong deblocking filtering if $2 \times (dp_i + dq_i) < (\beta \gg 2)$, $2 \times (dp_{i_side} + dq_{i_side}) < (\beta \gg 2)$ and $|p_{0i} - q_{0i}| < (t_c \ll 4)$.

If strong deblocking filtering is selected in step S61 the method continues to step S55 of Fig. 13 otherwise the method continues to step S56 of Fig. 13.

10

The values of the parameters β and t_c can, for instance, be obtained from Table 1 using the quantization parameter (QP) value of the block 10 of pixels 12 as input.

In the foregoing, the deblocking filter control method has been discussed applied to a line 15 of pixels 12, 22 across a block boundary 2. This line 15 of pixels 12, 22 has been denoted line number i . The decision to apply strong or weak deblocking filtering of embodiments could be applied to any line 15 of pixels 12, 22 in the block 10 of pixels 12. For instance, if the block 10 of pixels 12 comprises four lines 15 of pixels 12, 22 as shown in Fig. 17, i.e. $i=0\dots3$, the filter decision could be applied to any of these lines $i=0\dots3$. Alternatively, the selection between strong and weak deblocking filtering is only applicable to a subset of the lines 15 of pixels 12, 14. For instance, the filter decision could be applied to line number $i=0$ and line number $i=3$. In such a case, strong deblocking filtering is typically never applied and thereby not available for the remaining lines 15 of pixels 12, 22 in the block 10 of pixels 12, i.e. for line number $i=1$ and line number $i=2$. In yet another approach, the filter decision is only applied to one of the lines, for instance line number $i=0$, or line number $i=1$, or line number $i=2$ or line number $i=3$. In an embodiment adapted for HEVC the filter decision is applied to the line number $i=0$ and the line number $i=3$ and strong deblocking filtering is applied if the decision to apply strong deblocking filtering holds for both these lines.

30

In particular embodiments, the filtering decision is performed as described in the following.

Let the following values be defined as below

$$\begin{aligned} dp_{0_side} &= |p_{3_0} - 2 \times p_{2_0} + p_{1_0}| \\ dp_{3_side} &= |p_{3_3} - 2 \times p_{2_3} + p_{1_3}| \\ dq_{0_side} &= |q_{3_0} - 2 \times q_{2_0} + q_{1_0}| \end{aligned} \tag{9}$$

$$dq3_side = | q3_3 - 2 \times q2_3 + q1_3 |$$

and the values $dp0$, $dp3$, $dq0$, $dq3$, $dpq0$, $dpq3$, dp , dq are derived as in equations (1) in the background section.

- 5 First, a condition in equation (4) or a similar condition, which checks that the first three pixels 12, 22 from the block boundary 2 form a straight line, is checked.

In addition to that, the condition from equation (6) that the change between first two pixels 12, 22 closest to the block boundary 2 is below a certain threshold may additionally be evaluated.

10

A criterion for application of the strong filtering includes comparing the calculated values of at least $dp0_side$ and $dq0_side$ with the corresponding threshold value(s). Alternatively, a sum of values $dp0_side$ and $dq0_side$ can be used in a comparison.

- 15 The strong filtering decisions can be done based on one or more lines 15 across the block boundary 2. For example, the decisions based on lines 0 and 3 can be used. In this case, the decision to apply the strong filter is done based on fact that the strong deblocking filtering criteria for both line 0 and line 3 holds. Alternatively a sum of the values $dp0_side$, $dp3_side$, $dq0_side$ and $dq3_side$ can be used. The respective values for other lines 15 than 0 and 3 can be used. The strong filtering decision can be
- 20 performed separately for each line 15 or for a set of lines 15 across the block boundary 2 based on a subset of line 15 across the block boundary 2.

Figs. 1B and 1C are flow diagrams illustrating embodiments performed in a transmitter and a receiver, respectively. Fig. 1B involves sending signaling according to the embodiments and Fig. 1C

25 correspondingly involves receiving signaling according to the embodiments.

Embodiments

The embodiments below are the examples of the proposed strong filtering decisions. These examples are, however, not restrictive. For example, different values of thresholds can be used. Moreover, the

30 criteria for the strong filtering decision can also be used in combination with some other criteria for the strong filtering to be applied.

Embodiment 1

An example of the proposed strong filtering decisions are shown below

$$dp0 = | p2_0 - 2 \times p1_0 + p0_0 | \quad (1)$$

$$dp3 = | p2_3 - 2 \times p1_3 + p0_3 |$$

$$dq0 = | q2_0 - 2 \times q1_0 + q0_0 |$$

$$dq3 = | q2_3 - 2 \times q1_3 + q0_3 |$$

5

$$dp0_side = | p3_0 - 2 \times p2_0 + p1_0 | \quad (9)$$

$$dp3_side = | p3_3 - 2 \times p2_3 + p1_3 |$$

$$dq0_side = | q3_0 - 2 \times q2_0 + q1_0 |$$

$$dq3_side = | q3_3 - 2 \times q2_3 + q1_3 |$$

10

The strong filtering is applied if every of the following conditions holds:

$$dp0 + dq0 < Thr1$$

$$dp0_side + dq0_side < Thr2$$

$$| p0_0 - q0_0 | < Thr3$$

15

Here the values of Thr1, Thr2 and Thr3 depend on the quantization parameter (QP) and may be derived from the look-up tables.

Embodiment 2

20 Another embodiment of the proposed strong filter decision is shown below. Here the values of $dp0_side$ and $dq0_side$ are evaluated separately.

$$dp0 = | p2_0 - 2 \times p1_0 + p0_0 | \quad (1)$$

$$dp3 = | p2_3 - 2 \times p1_3 + p0_3 |$$

$$dq0 = | q2_0 - 2 \times q1_0 + q0_0 |$$

$$25 \quad dq3 = | q2_3 - 2 \times q1_3 + q0_3 |$$

$$dp0_side = | p3_0 - 2 \times p2_0 + p1_0 | \quad (9)$$

$$dp3_side = | p3_3 - 2 \times p2_3 + p1_3 |$$

$$dq0_side = | q3_0 - 2 \times q2_0 + q1_0 |$$

$$30 \quad dq3_side = | q3_3 - 2 \times q2_3 + q1_3 |$$

The strong filtering is applied if every of the following conditions holds:

$$dp0 + dq0 < Thr1$$

$$dp0_side < Thr2$$

$$\begin{aligned} dq0_side &< Thr2 \\ |p0_0 - q0_0| &< Thr3 \end{aligned}$$

Here the values of Thr1, Thr2 and Thr3 depend on the quantization parameter (QP) and may be
5 derived from the look-up tables.

Embodiment 3

An example of the proposed strong filter decision, where the conditions are evaluated separately is
given below.

$$\begin{aligned} 10 \quad dp0 &= |p2_0 - 2 \times p1_0 + p0_0| & (1) \\ dp3 &= |p2_3 - 2 \times p1_3 + p0_3| \\ dq0 &= |q2_0 - 2 \times q1_0 + q0_0| \\ dq3 &= |q2_3 - 2 \times q1_3 + q0_3| \end{aligned}$$

$$\begin{aligned} 15 \quad dp0_side &= |p3_0 - 2 \times p2_0 + p1_0| & (9) \\ dp3_side &= |p3_3 - 2 \times p2_3 + p1_3| \\ dq0_side &= |q3_0 - 2 \times q2_0 + q1_0| \\ dq3_side &= |q3_3 - 2 \times q2_3 + q1_3| \end{aligned}$$

20 The strong filtering is applied if every of the following conditions holds:

$$\begin{aligned} dp0 &< Thr1 \\ dq0 &< Thr1 \\ dp0_side &< Thr2 \\ dq0_side &< Thr2 \\ 25 \quad |p0_0 - q0_0| &< Thr3 \end{aligned}$$

Here the values of Thr1, Thr2 and Thr3 depend on the quantization parameter (QP) and may be
derived from the look-up tables.

30 Embodiment 4

An example of the proposed strong filter decision, the criteria can be used based on more than one
line, for example, line 0 and line 3 is shown below.

$$\begin{aligned} dp0 &= |p2_0 - 2 \times p1_0 + p0_0| & (1) \\ dp3 &= |p2_3 - 2 \times p1_3 + p0_3| \end{aligned}$$

$$dq0 = |q2_0 - 2 \times q1_0 + q0_0|$$

$$dq3 = |q2_3 - 2 \times q1_3 + q0_3|$$

$$dp0_side = |p3_0 - 2 \times p2_0 + p1_0| \tag{9}$$

$$5 \quad dp3_side = |p3_3 - 2 \times p2_3 + p1_3|$$

$$dq0_side = |q3_0 - 2 \times q2_0 + q1_0|$$

$$dq3_side = |q3_3 - 2 \times q2_3 + q1_3|$$

The strong filtering is applied if every of the following conditions holds:

$$10 \quad dp0 + dq0 < Thr1$$

$$dp0_side + dq0_side < Thr2$$

$$|p0_0 - q0_0| < Thr3$$

$$dp3 + dq3 < Thr1$$

$$15 \quad dp3_side + dq3_side < Thr2$$

$$|p0_3 - q0_3| < Thr3$$

Pixel values from some other lines than 0 and 3, e.g. line 1 and 2, can be used.

20 Here the values of Thr1, Thr2 and Thr3 depend on the quantization parameter (QP) and may be derived from the look-up tables.

Embodiment 5

In this embodiment, a check for the maximum allowed distance between pixels closest to the block

25 boundary is not used.

$$dp0 = |p2_0 - 2 \times p1_0 + p0_0| \tag{1}$$

$$dp3 = |p2_3 - 2 \times p1_3 + p0_3|$$

$$dq0 = |q2_0 - 2 \times q1_0 + q0_0|$$

$$dq3 = |q2_3 - 2 \times q1_3 + q0_3|$$

30

$$dp0_side = |p3_0 - 2 \times p2_0 + p1_0| \tag{9}$$

$$dp3_side = |p3_3 - 2 \times p2_3 + p1_3|$$

$$dq0_side = |q3_0 - 2 \times q2_0 + q1_0|$$

$$dq3_side = |q3_3 - 2 \times q2_3 + q1_3|$$

The strong filtering is applied if every of the following conditions holds:

$$dp0+dq0 < Thr1$$

$$dp0_side+dq0_side < Thr2$$

$$5 \quad dp3+dq3 < Thr1$$

$$dp3_side+dq3_side < Thr2$$

Pixel values from some other lines than 0 and 3, e.g. line 1 and 2, can be used.

- 10 Here the values of Thr1 and Thr2 depend on the quantization parameter (QP) and may be derived from the look-up tables.

Embodiment 6

An example of the proposed strong filter decision with the values of the thresholds is given.

$$15 \quad dp0 = | p2_0 - 2 \times p1_0 + p0_0 | \quad (1)$$

$$dp3 = | p2_3 - 2 \times p1_3 + p0_3 |$$

$$dq0 = | q2_0 - 2 \times q1_0 + q0_0 |$$

$$dq3 = | q2_3 - 2 \times q1_3 + q0_3 |$$

$$20 \quad dp0_side = | p3_0 - 2 \times p2_0 + p1_0 | \quad (9)$$

$$dp3_side = | p3_3 - 2 \times p2_3 + p1_3 |$$

$$dq0_side = | q3_0 - 2 \times q2_0 + q1_0 |$$

$$dq3_side = | q3_3 - 2 \times q2_3 + q1_3 |$$

- 25 The strong filtering is applied if every of the following conditions holds:

$$2 \times (dp0+dq0) < (\text{beta} \gg 2)$$

$$2 \times (dp0_side+dq0_side) < (\text{beta} \gg 2)$$

$$|p0_0 - q0_0| < (\text{tc} \ll 4)$$

$$30 \quad 2 \times (dp3+dq3) < (\text{beta} \gg 2)$$

$$2 \times (dp3_side+dq3_side) < (\text{beta} \gg 2)$$

$$|p0_3 - q0_3| < (\text{tc} \ll 4)$$

This is equivalent to:

$$\begin{aligned}
& 2 \times (dp0 + dq0) < (\text{beta} \gg 2) \\
& ((dp0_side + dq0_side) \ll 1) < (\text{beta} \gg 2) \\
& |p0_0 - q0_0| < (tc \ll 4)
\end{aligned}$$

$$\begin{aligned}
5 \quad & 2 \times (dp3 + dq3) < (\text{beta} \gg 2) \\
& ((dp3_side + dq3_side) \ll 1) < (\text{beta} \gg 2) \\
& |p0_3 - q0_3| < (tc \ll 4)
\end{aligned}$$

Pixel values from some other lines than 0 and 3, e.g. line 1 and 2, can be used.

10

Here the values of beta and tc depend on the quantization parameter (QP) and may be derived from the look-up tables.

Embodiment 7

15 An example of the proposed strong filter decisions with the values of the thresholds is given.

$$\begin{aligned}
dp0 &= |p2_0 - 2 \times p1_0 + p0_0| & (1) \\
dp3 &= |p2_3 - 2 \times p1_3 + p0_3| \\
dq0 &= |q2_0 - 2 \times q1_0 + q0_0| \\
dq3 &= |q2_3 - 2 \times q1_3 + q0_3|
\end{aligned}$$

20

$$\begin{aligned}
dp0_side &= |p3_0 - 2 \times p2_0 + p1_0| & (9) \\
dp3_side &= |p3_3 - 2 \times p2_3 + p1_3| \\
dq0_side &= |q3_0 - 2 \times q2_0 + q1_0| \\
dq3_side &= |q3_3 - 2 \times q2_3 + q1_3|
\end{aligned}$$

25

The strong filtering is applied if every of the following conditions holds:

$$\begin{aligned}
& dp0 + dq0 < (\text{beta} \gg 3) \\
& dp0_side + dq0_side < (\text{beta} \gg 3) \\
& |p0_0 - q0_0| < (tc \ll 4)
\end{aligned}$$

30

$$\begin{aligned}
& (dp3 + dq3) < (\text{beta} \gg 3) \\
& (dp3_side + dq3_side) < (\text{beta} \gg 3) \\
& |p0_3 - q0_3| < (tc \ll 4)
\end{aligned}$$

Pixel values from some other lines than 0 and 3, e.g. line 1 and 2, can be used.

Here the values of beta and tc depend on the quantization parameter (QP) and may be derived from the look-up tables.

5

Embodiment 8

An example of the proposed strong filter decision with suggested values of the thresholds is given.

$$dp0 = | p2_0 - 2 \times p1_0 + p0_0 | \quad (1)$$

$$dp3 = | p2_3 - 2 \times p1_3 + p0_3 |$$

10
$$dq0 = | q2_0 - 2 \times q1_0 + q0_0 |$$

$$dq3 = | q2_3 - 2 \times q1_3 + q0_3 |$$

$$dp0_side = | p3_0 - 2 \times p2_0 + p1_0 | \quad (9)$$

$$dp3_side = | p3_3 - 2 \times p2_3 + p1_3 |$$

15
$$dq0_side = | q3_0 - 2 \times q2_0 + q1_0 |$$

$$dq3_side = | q3_3 - 2 \times q2_3 + q1_3 |$$

The strong filtering is applied if every of the following conditions holds:

$$dp0 + dq0 < (\text{beta} \gg 3)$$

20
$$dp0_side < (\text{beta} \gg 4)$$

$$dq0_side < (\text{beta} \gg 4)$$

$$| p0_0 - q0_0 | < (\text{tc} \ll 4)$$

$$(dp3 + dq3) < (\text{beta} \gg 3)$$

25
$$dp3_side < (\text{beta} \gg 4)$$

$$dq3_side < (\text{beta} \gg 4)$$

$$| p0_3 - q0_3 | < (\text{tc} \ll 4)$$

Pixel values from some other lines than 0 and 3, e.g. line 1 and 2, can be used.

30

Here the values of beta and tc depend on the quantization parameter (QP) and may be derived from the look-up tables.

Embodiment 9

An example of the proposed strong filter decision with suggested values of the thresholds is given.

$$dp0 = | p2_0 - 2 \times p1_0 + p0_0 | \quad (1)$$

$$dp3 = | p2_3 - 2 \times p1_3 + p0_3 |$$

$$dq0 = | q2_0 - 2 \times q1_0 + q0_0 |$$

$$5 \quad dq3 = | q2_3 - 2 \times q1_3 + q0_3 |$$

$$dp0_side = | p3_0 - 2 \times p2_0 + p1_0 | \quad (9)$$

$$dp3_side = | p3_3 - 2 \times p2_3 + p1_3 |$$

$$dq0_side = | q3_0 - 2 \times q2_0 + q1_0 |$$

$$10 \quad dq3_side = | q3_3 - 2 \times q2_3 + q1_3 |$$

The strong filtering is applied if every of the following conditions holds:

$$dp0 + dq0 + dp3 + dq3 < Thr1$$

$$dp0_side + dq0_side + dp3_side + dq3_side < Thr2$$

$$15 \quad |p0_0 - q0_0| < Thr3$$

$$|p0_3 - q0_3| < Thr3$$

Pixel values from some other lines than 0 and 3, e.g. line 1 and 2, can be used.

20 Here the values of Thr1, Thr2 and Thr3 depend on the quantization parameter (QP) and may be derived from the look-up tables.

Embodiment 10

An example of the proposed strong filter decision with suggested values of the thresholds is given.

$$25 \quad dp0 = | p2_0 - 2 \times p1_0 + p0_0 | \quad (1)$$

$$dp3 = | p2_3 - 2 \times p1_3 + p0_3 |$$

$$dq0 = | q2_0 - 2 \times q1_0 + q0_0 |$$

$$dq3 = | q2_3 - 2 \times q1_3 + q0_3 |$$

$$30 \quad dp0_side = | p3_0 - 2 \times p2_0 + p1_0 | \quad (9)$$

$$dp3_side = | p3_3 - 2 \times p2_3 + p1_3 |$$

$$dq0_side = | q3_0 - 2 \times q2_0 + q1_0 |$$

$$dq3_side = | q3_3 - 2 \times q2_3 + q1_3 |$$

The strong filtering is applied if every of the following conditions holds:

$$dp_0 + dq_0 + dp_3 + dq_3 < \text{Thr1}$$

$$dp_{0_side} + dq_{0_side} + dp_{3_side} + dq_{3_side} < \text{Thr 2}$$

$$|p_{0_0} - q_{0_0}| + |p_{0_3} - q_{0_3}| < \text{Thr3}$$

5

Pixel values from some other lines than 0 and 3, e.g. line 1 and 2, can be used.

Here the values of Thr1, Thr2 and Thr3 depend on the quantization parameter (QP) and may be derived from the look-up tables.

10

Embodiment 11

An example of the proposed strong filtering decisions are shown below

$$dpi = |p_{2_i} - 2 \times p_{1_i} + p_{0_i}|$$

$$dqj = |q_{2_i} - 2 \times q_{1_i} + q_{0_i}|$$

15

$$dpi_side = |p_{3_i} - 2 \times p_{2_i} + p_{1_i}|$$

$$dqj_side = |q_{3_i} - 2 \times q_{2_i} + q_{1_i}|$$

20 The strong filtering is applied to line number i if every of the following conditions holds:

$$dpi + dqj < \text{Thr1}$$

$$dpi_side + dqj_side < \text{Thr2}$$

$$|p_{0_i} - q_{0_i}| < \text{Thr 3}$$

25 Here the values of Thr1, Thr2 and Thr3 depend on the quantization parameter (QP) and may be derived from the look-up tables. The line could be line number $i=0$, line number $i=1$, line number $i=2$ or line number $i=3$.

Embodiment 12

30 Alternative embodiments of the above mentioned embodiments can use the same criteria but without comparisons with $|p_{0_0} - q_{0_0}|$ and $|p_{0_3} - q_{0_3}|$ to the threshold values. These embodiments can be obtained from the mentioned embodiments by removing the corresponding conditions.

The proposed embodiments improve the subjective and/or objective quality of the HEVC strong filtering. The embodiments enable application of the strong deblocking filter to the areas where the signal in the picture has a form of a gradient or ramp. It also enables application of the deblocking filter to the boundaries between heavily quantized blocks where the signal on the side of the block boundary is not flat. This enables application of the strong deblocking filtering to the areas where the blocking artifacts might otherwise be visible. The proposed filtering decisions can also be used with the strong deblocking filter that has ramp preservation properties. This would also enable better subjective quality of the image or video by more efficient reduction of blocking artifacts.

- 10 Fig. 7 is a schematic block diagram of a filtering control (ctrl) device 100 according to an embodiment. The filtering control device 100 comprises a determining unit 110 configured to determine or check whether pixel values of four pixels in a line of pixels in a block of pixels form an approximate line. The determining unit 110 is also configured to determine or check whether pixel values of four pixels in the line of pixels in a neighboring block of pixels form an approximate line. The filtering control device 100
15 also comprises a processing unit 120 configured to select to apply strong deblocking filtering to pixel values in the line of pixels in the block of pixels and the neighboring block of pixels if the pixel values of the four pixels in the block of pixels form an approximate line and if the pixel values of the four pixels in the neighboring block of pixels form an approximate line.
- 20 The determining unit 110 of the filtering control device 100 is also referred to herein as a determiner or determining means or module. Correspondingly, the processing unit 120 is also referred to as processor or processing means or module.

In an embodiment, the processing unit 120 is configured to select to apply weak deblocking filtering to
25 pixel values in the line of pixels in the block of pixels and the neighboring block of pixels if the pixel values of the four pixels in the block of pixels do not form an approximate line and/or if the pixel values of the four pixels in the neighboring block of pixels do not form an approximate line.

The determining unit 110 is preferably configured, in an embodiment, to determine whether the pixel
30 values of the four pixels in the line of pixels in the block of pixels form a flat line or a ramp. The determining unit 110 is preferably also configured to determine whether the pixel values of the four pixels in the line of pixels in the neighboring block of pixels form a flat line or a ramp. In this embodiment, the processing unit 120 is preferably configured to select to apply strong deblocking

filtering if the pixel values of the four pixels in the block of pixels form a flat line or a ramp and if the pixel values of the four pixels in the neighboring block of pixels form a flat ramp or a ramp.

The filtering control device 100 may additionally use an additional criterion in order to select between strong and weak deblocking filtering. In such a case, the determining unit 110 is preferably configured to determine whether a difference between i) a pixel value of a first pixel, relative to the block boundary, in the line of pixels in the block of pixels and ii) a pixel value or a first pixel, relative to the block boundary, in the line of pixels in the neighboring block of pixels is smaller than a threshold value. The processing unit 120 is then preferably configured to select to apply strong deblocking filtering if the pixel values of the four pixels in the block of pixels form an approximate line, if the pixel values of the four pixels in the neighboring block of pixels form an approximate line and if the difference is smaller than the threshold value. If at least one of these criteria is not met the processing unit preferably selects to apply weak deblocking filtering for the pixel values in the current line of pixels.

In a particular embodiment, the determining unit 110 is configured to determine whether pixel values of a first pixel, a second pixel and a third pixel, relative to the block boundary, in the line of pixels in the block of pixels form an approximate line. The determining unit 110 is also configured to determine whether pixel values of the second pixel, the third pixel and a fourth pixel, relative to the block boundary, in the line of pixels in the block of pixels form an approximate line. In this embodiment the determining unit 110 is further configured to determine whether pixel values of a first pixel, a second pixel and a third pixel, relative to the block boundary, in the line of pixels in the neighboring block of pixels form an approximate line. The determining unit 110 is additionally configured to determine whether pixel values of the second pixel, the third pixel and a fourth pixel, relative to the block boundary, in the line of pixels in the neighboring block of pixels form an approximate line.

25

The processing unit 120 is, in this embodiment, preferably configured to select to apply strong deblocking filtering if the pixel values of the first pixel, the second pixel and the third pixel in the block of pixels form an approximate line, if the pixel values of the second pixel, the third pixel and the fourth pixel in the block of pixels form an approximate line, if the pixel values of the first pixel, the second pixel and the third pixel in the neighboring block of pixels form an approximate line and if the pixel values of the second pixel, the third pixel and the fourth pixel in the neighboring block of pixels form an approximate line.

30

Fig. 15 illustrates an implementation example of the determining unit 110. The determining unit 110 then comprises a dp_i calculator 111 configured to calculate $dp_i = |p_2 - 2 \times p_1 + p_0|$. The determining unit 110 also comprises a dp_{i_side} calculator 113 configured to calculate $dp_{i_side} = |p_3 - 2 \times p_2 + p_1|$. A dq_i calculator 115 is provided in the determining unit 110 and is configured to calculate $dq_i = |q_2 - 2 \times q_1 + q_0|$. The determining unit 110 further comprises a dq_{i_side} calculator 117 configured to calculate $dq_{i_side} = |q_3 - 2 \times q_2 + q_1|$. In this implementation example the processing unit 120 of the filtering control device 100 is configured to select to apply strong deblocking filtering if $(dp_i + dq_i) < Thr1$ and $(dp_{i_side} + dq_{i_side}) < Thr2$.

10 In a particular embodiment of this implementation example the processing unit 120 is configured to select to apply strong deblocking filtering if $2 \times (dp_i + dq_i) < (\beta \gg 2)$ and $2 \times (dp_{i_side} + dq_{i_side}) < (\beta \gg 2)$.

In another implementation example, the determining unit 110 comprises the previously mentioned dp_i calculator 111, dp_{i_side} calculator 113, dq_i calculator 115 and dq_{i_side} calculator 117. The determining unit 110 preferably also comprises a $|p_0 - q_0|$ calculator 119 configured to calculate $|p_0 - q_0|$. In this another implementation example the processing unit 120 is configured to select to apply strong deblocking filtering if $(dp_i + dq_i) < Thr1$, $(dp_{i_side} + dq_{i_side}) < Thr2$ and $|p_0 - q_0| < Thr3$.

20 In a particular embodiment of this another implementation example the processing unit 120 is configured to select to apply strong deblocking filtering if $2 \times (dp_i + dq_i) < (\beta \gg 2)$, $2 \times (dp_{i_side} + dq_{i_side}) < (\beta \gg 2)$ and $|p_0 - q_0| < ((5 \times t_c + 1) \gg 1)$.

In another embodiment of this another implementation example the processing unit 120 is configured to select to apply strong deblocking filtering if $2 \times (dp_i + dq_i) < (\beta \gg 2)$, $2 \times (dp_{i_side} + dq_{i_side}) < (\beta \gg 2)$ and $|p_0 - q_0| < (t_c \ll 4)$.

Accordingly as illustrated in Fig. 7, the filtering control device 100 implements the functions of embodiments 1-12 or a combination thereof by the determining unit 110 and the processing unit 120, which are configured to determine if the signal on both sides of the block boundary is smooth.

The filtering control device 100 of Fig. 7 with its including units 110-120 (and optional units 111-119 of the determining unit 110 in Fig. 15) could be implemented in hardware. There are numerous variants of circuitry elements that can be used and combined to achieve the functions of the units 110-120 of the filtering

control device 100. Such variants are encompassed by the embodiments. Particular examples of hardware implementation of the filtering control device 100 is implementation in digital signal processor (DSP) hardware and integrated circuit technology, including both general-purpose electronic circuitry and application-specific circuitry.

5

Hence, the steps, functions, procedures and/or blocks described above may be implemented in hardware using any conventional technology, such as discrete circuit or integrated circuit technology, including both general-purpose electronic circuitry and application-specific circuitry.

10 Particular examples include one or more suitably configured digital signal processors and other known electronic circuits, e.g. discrete logic gates interconnected to perform a specialized function, or Application Specific Integrated Circuits (ASICs).

Alternatively, at least some of the steps, functions, procedures and/or blocks described above may be
15 implemented in software such as a computer program for execution by suitable processing circuitry including one or more processing units. Examples of processing circuitry includes, but is not limited to, one or more microprocessors, one or more DSPs, one or more Central Processing Units (CPUs), video acceleration hardware, and/or any suitable programmable logic circuitry such as one or more Field Programmable Gate Arrays (FPGAs) or one or more Programmable Logic Controllers (PLCs).

20

It should also be understood that it may be possible to re-use the general processing capabilities of any conventional device or unit in which the proposed technology is implemented. It may also be possible to re-use existing software, e.g. by reprogramming of the existing software or by adding new software components.

25

The filtering control device 100 described herein could alternatively be implemented e.g. by one or more of a processing unit 72 in a computer 70 and adequate software with suitable storage or memory therefore, a programmable logic device (PLD) or other electronic component(s) as shown in Fig. 8.

30 In the following, an example of a computer implementation will be described with reference to Fig. 8. The computer 70 comprises a processing unit 72, such as one or more processors or a processing circuit, and a memory 73 represented as a computer program product in the figure. In this particular example, at least some of the steps, functions, procedures and/or blocks described above are implemented in a computer program 74, which is loaded into the memory 73 for execution by the

processing unit 72. The processing unit 72 and memory 73 are interconnected to each other to enable normal software execution. An optional input/output unit 71 may also be interconnected to the processing unit 72 and/or the memory 73 to enable input and/or output of relevant data such as input parameter(s) and/or resulting output parameter(s).

5

The term 'computer' should be interpreted in a general sense as any system or device capable of executing program code or computer program instructions to perform a particular processing, determining or computing task.

10 The software or computer program 74 may be realized as a computer program product 73, which is normally carried or stored on a computer-readable medium. The computer-readable medium may include one or more removable or non-removable memory devices including, but not limited to a Read-Only Memory (ROM), a Random Access Memory (RAM), a Compact Disc (CD), a Digital Versatile Disc (DVD), a Universal Serial Bus (USB) memory, a Hard Disk Drive (HDD) storage device, a flash
15 memory, or any other conventional memory device. The computer program 74 may thus be loaded into the operating memory of a computer 70 or equivalent processing device for execution by the processing unit 72 thereof.

For example, the computer program 74 stored in memory includes program instructions executable by
20 the processing unit 72, whereby the processing unit 72 is able or operative to execute the above-described steps, functions, procedure and/or blocks.

The computer 70 or processing unit 72 does not have to be dedicated to only execute the above-described steps, functions, procedure and/or blocks, but may also execute other tasks.

25

Fig. 8 schematically illustrates an embodiment of a computer 70 having a processing unit 72, such as a DSP or CPU. The processing unit 72 can be a single unit or a plurality of units for performing different steps of the method described herein. The computer 70 also comprises an optional input/output (I/O) unit 71 for receiving recorded or generated video frames or encoded video frames and outputting
30 encoded video frame or decoded video data. The I/O unit 71 has been illustrated as a single unit in Fig. 8 but can likewise be in the form of a separate input unit and a separate output unit.

Furthermore, the computer 70 comprises at least one computer program product 73 in the form of a non-volatile memory, for instance an EEPROM (Electrically Erasable Programmable Read-Only

Memory) or a flash memory. The computer program product 73 comprises a computer program 74, which comprises code means 310 which when run on or executed by the computer 70, such as by the processing unit 72, causes the computer 70 to perform the steps of the method described in the foregoing in connection with Figs. 1, 9-14. Hence, in an embodiment the code means 310 in the
5 computer program 74 comprises a module configured to implement embodiments 1-12 or combinations thereof. This module essentially performs the steps of the flow diagrams in Figs. 1, 9-14 when run on the processing unit 72. Thus, when the module is run on the processing unit 72 they correspond to the corresponding units 110-120 of Fig 7.

10 An embodiment therefore relates to a computer 70 comprising a processing unit 72 and a computer program product 73 comprising a computer program 74. The computer program 74 comprises code means 310, which when run on the processing unit 72, causes the processing unit 72 to check whether pixel values of four pixels in a line of pixels in a block of pixels form an approximate line. The computer program 74 also causes the processing unit 72 to check whether pixel values of four pixels in the line
15 of pixels in a neighboring block of pixels form an approximate line. The computer program 74 further causes the processing unit 72 to select to apply strong deblocking filtering to pixel values in the line of pixels in the block of pixels and the neighboring block of pixels if the pixel values of the four pixels in the block of pixels form an approximate line and if the pixel values of the four pixels in the neighboring block of pixels form an approximate line.

20

An embodiment also relates to a computer program 74 for filtering control. The computer program 74 comprises code means 310, which when executed by the processing unit 72, causes the processing unit 72 to check whether pixel values of four pixels in a line of pixels in a block of pixels form an approximate line. The computer program 74 also causes the processing unit 72 to check whether pixel
25 values of four pixels in the line of pixels in a neighboring block of pixels form an approximate line. The computer program 74 further causes the processing unit 72 to select to apply strong deblocking filtering to pixel values in the line of pixels in the block of pixels and the neighboring block of pixels if the pixel values of the four pixels in the block of pixels form an approximate line and if the pixel values of the four pixels in the neighboring block of pixels form an approximate line.

30

An embodiment further relates to a computer program product 73 comprising computer readable code means and a computer program 74 according to above stored on the computer readable code means.

The methods of the embodiments are performed in a filtering control device which can be located in an encoder or a decoder as schematically illustrated in Figs. 2 and 3. Figs. 2 and 3 illustrate the example when the method is performed inside the coding loop.

5 The embodiments can be implemented in a filtering control device.

Fig. 2 is a schematic block diagram of an encoder 40 for encoding a block of pixels in a video frame of a video sequence according to an embodiment.

10 A current block of pixels is predicted by performing a motion estimation by a motion estimator 50 from an already provided block of pixels in the same frame or in a previous frame. The result of the motion estimation is a motion or displacement vector associated with the reference block, in the case of inter prediction. The motion vector is utilized by a motion compensator 50 for outputting an inter prediction of the block of pixels.

15

An intra predictor 49 computes an intra prediction of the current block of pixels. The outputs from the motion estimator/compensator 50 and the intra predictor 49 are input in a selector 51 that either selects intra prediction or inter prediction for the current block of pixels. The output from the selector 51 is input to an error calculator in the form of an adder 41 that also receives the pixel values of the current block
20 of pixels. The adder 41 calculates and outputs a residual error as the difference in pixel values between the block of pixels and its prediction.

The error is transformed in a transformer 42, such as by a discrete cosine transform, and quantized by a quantizer 43 followed by coding in an encoder 44, such as by entropy encoder. In inter coding, also
25 the estimated motion vector is brought to the encoder 44 for generating the coded representation of the current block of pixels.

The transformed and quantized residual error for the current block of pixels is also provided to an inverse quantizer 45 and inverse transformer 46 to retrieve the original residual error. This error is
30 added by an adder 47 to the block prediction output from the motion compensator 50 or the intra predictor 49 to create a reference block of pixels that can be used in the prediction and coding of a next block of pixels. This new reference block is first processed by a filtering control device 100 in order to control any filtering that is applied to the reference block to combat any artifact. The processed new

reference block is then temporarily stored in a frame buffer 48, where it is available to the intra predictor 49 and the motion estimator/compensator 50.

Fig. 3 is a corresponding schematic block diagram of a decoder 60 comprising a filtering control device 100 according to any of the embodiments or in combinations thereof. The decoder 60 comprises a decoder 61, such as entropy decoder, for decoding an encoded representation of a block of pixels to get a set of quantized and transformed residual errors. These residual errors are dequantized in an inverse quantizer 62 and inverse transformed by an inverse transformer 63 to get a set of residual errors.

10

These residual errors are added in an adder 64 to the pixel values of a reference block of pixels. The reference block is determined by a motion estimator/compensator 67 or intra predictor 66, depending on whether inter or intra prediction is performed. A selector 68 is thereby interconnected to the adder 64 and the motion estimator/compensator 67 and the intra predictor 66. The resulting decoded block of pixels output from the adder 64 is input to a filtering control device 100 in order to control any filter that is applied to combat any artifacts. The filtered block of pixels is output from the decoder 60 and is furthermore preferably temporarily provided to a frame buffer 65 and can be used as a reference block of pixels for a subsequent block of pixels to be decoded. The frame buffer 65 is thereby connected to the motion estimator/compensator 67 to make the stored blocks of pixels available to the motion estimator/compensator 67.

20

The output from the adder 64 is preferably also input to the intra predictor 66 to be used as an unfiltered reference block of pixels.

In the embodiments disclosed in Figs. 2 and 3 the filtering control device 100 controls filtering in the form of so called in-loop filtering. In an alternative implementation at the decoder 60 the filtering control device 100 is arranged to perform so called post-processing filtering. In such a case, the filtering control device 100 operates on the output frames outside of the loop formed by the adder 64, the frame buffer 65, the intra predictor 66, the motion estimator/compensator 67 and the selector 68. No filtering and filter control is then typically done at the encoder.

30

The solution, including the embodiments above, applies to a decoder, an encoder, as an in-loop filter or as a post-processing filter. The encoder may for example be located in a transmitter in a video camera

in e.g. a mobile device. The decoder may for example be located in a receiver in a video camera or any other device for displaying, decoding or transcoding a video stream.

The solution is not limited to HEVC but may be applied to any extension of HEVC such as a scalable
5 extension or multiview extension or to a different video codec.

Fig. 5 is a schematic block diagram of a user equipment or media terminal 80 housing a decoder 60 with a filtering control device. The user equipment 80 can be any device having media decoding functions that operates on an encoded video stream of encoded video frames to thereby decode the
10 video frames and make the video data available. Non-limiting examples of such devices include mobile telephones and other portable media players, tablets, desktops, notebooks, personal video recorders, multimedia players, video streaming servers, set-top boxes, TVs, computers, decoders, game consoles, etc. The user equipment 80 comprises a memory 84 configured to store encoded video frames. These encoded video frames can have been generated by the user equipment 80 itself.
15 Alternatively, the encoded video frames are generated by some other device and wirelessly transmitted or transmitted by wire to the user equipment 80. The user equipment 80 then comprises a transceiver (transmitter and receiver) or input and output port 82 to achieve the data transfer.

The encoded video frames are brought from the memory 84 to a decoder 60, such as the decoder
20 illustrated in Fig. 3. The decoder 60 comprises a filtering control device according one of embodiments 1-12 or combinations thereof. The decoder 60 then decodes the encoded video frames into decoded video frames. The decoded video frames are provided to a media player 86 that is configured to render the decoded video frames into video data that is displayable on a display or screen 88 of or connected to the user equipment 80.

25

In Fig. 5, the user equipment 80 has been illustrated as comprising both the decoder 60 and the media player 86, with the decoder 60 implemented as a part of the media player 86. This should, however, merely be seen as an illustrative but non-limiting example of an implementation embodiment for the user equipment 80. Also distributed implementations are possible where the decoder 60 and the media
30 player 86 are provided in two physically separated devices are possible and within the scope of user equipment 80 as used herein. The display 88 could also be provided as a separate device connected to the user equipment 80, where the actual data processing is taking place.

Fig. 4 illustrates another embodiment of a user equipment 80 that comprises an encoder, such as the encoder of Fig. 2, comprising a filtering control device according to the embodiments. The encoder 40 is then configured to encode video frames received by the I/O unit 82 and/or generated by the user equipment 80 itself. In the latter case, the user equipment 80 preferably comprises a media engine or recorder, such as in the form of or connected to a (video) camera. The user equipment 80 may optionally also comprise a media player 86, such as a media player 86 with a decoder and filtering control device according to the embodiments, and a display 88.

As illustrated in Fig. 6, the encoder 40 and/or decoder 60, such as illustrated in Figs. 2 and 3, may be implemented in a network device 30 being or belonging to a network node in a communication network 32 between a sending unit 34 and a receiving user equipment 36. Such a network device 30 may be a device for converting video according to one video coding standard to another video coding standard, for example, if it has been established that the receiving user equipment 36 is only capable of or prefers another video coding standard than the one sent from the sending unit 34. The network device 30 can be in the form of or comprised in a radio base station, a Node-B or any other network node in a communication network 32, such as a radio-based network.

The embodiments described above are merely given as examples, and it should be understood that the proposed technology is not limited thereto. It will be understood by those skilled in the art that various modifications, combinations and changes may be made to the embodiments without departing from the present scope as defined by the appended claims. In particular, different part solutions in the different embodiments can be combined in other configurations, where technically possible.

CLAIMS

1. A deblocking filter control method comprising:
 - a) checking (S10) whether pixels values of four pixels (12) in a line (15) of pixels (12, 22) in a block (10) of pixels (12) form an approximate line;
 - 5 b) checking (S11) whether pixel values of four pixels (22) in said line (15) of pixels (12, 22) in a neighboring block (20) of pixels (22) form an approximate line; and
selecting (S12) to apply strong deblocking filtering to pixel values in said line (15) of pixels (12, 22) in said block (10) of pixels (12) and said neighboring block (20) of pixels (22) if said pixel values of said four pixels (12) in said block (10) of pixels (12) form an approximate line and if said pixel values of
10 said four pixels (22) in said neighboring block (20) of pixels (22) form an approximate line.

2. The method according to claim 1, wherein selecting (S12) to apply strong deblocking filtering comprises selecting (S20) to apply weak deblocking filtering to pixel values in said line (15) of pixels (12, 22) in said block (10) of pixels (12) and said neighboring block (20) of pixels (22) if said pixel
15 values of said four pixels (12) in said block (10) of pixels (12) do not form an approximate line and/or if said pixel values of said four pixels (22) in said neighboring block (20) of pixels (22) do not form an approximate line.

3. The method according to claim 1 or 2, wherein
 - 20 a) checking (S10) whether said pixel values form an approximate line comprises checking (S11) whether said pixels values of said four pixels (12) in said line (15) of pixels (12, 22) in said block (10) of pixels (12) form a flat line or a ramp;
 - b) checking (S11) whether said pixel values form an approximate line comprises checking (S11) whether said pixel values of said four pixels (22) in said line (15) of pixels (12, 22) in said neighboring
25 block (20) of pixels (22) form a flat line or a ramp; and
selecting (S12) to apply strong deblocking filtering comprises selecting (S12) to apply strong deblocking filtering if said pixel values of said four pixels (12) in said block (10) of pixels (12) form a flat line or a ramp and if said pixel values of said four pixels (22) in said neighboring block (20) of pixels (22) form a flat line or a ramp.

- 30 4. The method according to any of the claims 1 to 3, further comprising checking (S30) whether a difference between i) a pixel value of a first pixel (12), relative to a block boundary (2) between said block (10) of pixels (12) and said neighboring block (20) of pixels (22), in said line (15) of pixels (12, 22) in said block (10) of pixels (12) and ii) a pixel value of a first pixel (22), relative to said block

boundary (2), in said line (15) of pixels (12, 22) in said neighboring block (20) of pixels (22) is smaller than a threshold value, wherein selecting (S12) to apply strong deblocking filtering comprises selecting (S12) to apply strong deblocking filtering if said pixel values of said four pixels (12) in said block (10) of pixels (12) form an approximate line, if said pixel values of said four pixels (22) in said neighboring block (20) of pixels (22) form an approximate line and if said difference is smaller than said threshold value.

5. The method according to any of the claims 1 to 4, wherein

a) checking (S10) whether said pixel values form an approximate line comprises:

10 a1) checking (S40) whether pixel values of a first pixel (12), a second pixel and a third pixel, relative to a block boundary (2) between said block (10) of pixels (12) and said neighboring block (20) of pixels (22), in said line (15) of pixels (12, 22) in said block (10) of pixels (12) form an approximate line; and

a2) checking (S41) whether pixel values of said second pixel, said third pixel and a fourth pixel, relative to said block boundary (2), in said line (15) of pixels (12, 22) in said block (10) of pixels (12) form an approximate line;

b) checking (S11) whether said pixel values form an approximate line comprises:

b1) checking (S42) whether pixel values of a first pixel (22), a second pixel and a third pixel, relative to said block boundary (2), in said line (15) of pixels (10, 12) in said neighboring block 20 (20) of pixels (22) form an approximate line; and

b2) checking (S43) whether pixel values of said second pixel, said third pixel and a fourth pixel, relative to said block boundary (2), in said line (15) of pixels (10, 12) in said neighboring block (20) of pixels (22) form an approximate line; and

selecting (S12) to apply strong deblocking filtering comprises selecting (S12) to apply strong 25 deblocking filtering if said pixel values of said first pixel (12), said second pixel and said third pixel in said block (10) of pixels (12) form an approximate line, if said pixel values of said second pixel, said third pixel and said fourth pixel in said block (10) of pixels (12) form an approximate line, if said pixel values of said first pixel (22), said second pixel and said third pixel in said neighboring block (20) of pixels (22) form an approximate line, and if said pixel values of said second pixel, said third pixel and 30 said fourth pixel in said neighboring block (20) of pixels (22) form an approximate line.

6. The method according to claim 5, wherein

a) checking (S10) whether said pixel values form an approximate line comprises:

a1) calculating (S50) $dp_i = | p_{2i} - 2 \times p_{1i} + p_{0i} |$, wherein p_{0i} denotes a pixel value of said first pixel (12) in said line (15) of pixels (12, 22) in said block (10) of pixels (12), p_{1i} denotes a pixel value of said second pixel in said line (15) of pixels (12, 22) in said block (10) of pixels (12), and p_{2i} denotes a pixel value of said third pixel in said line (15) of pixels (12, 22) in said block (10) of pixels (12); and

a2) calculating (S51) $dp_{i_side} = | p_{3i} - 2 \times p_{2i} + p_{1i} |$, wherein p_{3i} denotes a pixel value of said fourth pixel in said line (15) of pixels (12, 22) in said block (10) of pixels (12);

b) checking (S11) whether said pixel values form an approximate line comprises:

b1) calculating (S52) $dq_i = | q_{2i} - 2 \times q_{1i} + q_{0i} |$, wherein q_{0i} denotes a pixel value of said first pixel (22) in said line (15) of pixels (12, 22) in said neighboring block (20) of pixels (22), q_{1i} denotes a pixel value of said second pixel in said line (15) of pixels (12, 22) in said neighboring block (20) of pixels (22), and q_{2i} denotes a pixel value of said third pixel in said line (15) of pixels (12, 22) in said neighboring block (20) of pixels (22); and

b2) calculating (S53) $dq_{i_side} = | q_{3i} - 2 \times q_{2i} + q_{1i} |$, wherein q_{3i} denotes a pixel value of said fourth pixel in said line (15) of pixels (12, 22) in said neighboring block (20) of pixels (22); and

selecting (S12) to apply strong deblocking filtering comprises selecting (S54) to apply strong deblocking filtering if $(dp_i + dq_i) < Thr1$ and $(dp_{i_side} + dq_{i_side}) < Thr2$, wherein $Thr1$ denotes a first threshold value and $Thr2$ denotes a second threshold value.

7. The method according to claim 6, wherein selecting (S52) to apply strong deblocking filtering comprises selecting (S52) to apply strong deblocking filtering if $2 \times (dp_i + dq_i) < (\beta \gg 2)$ and $2 \times (dp_{i_side} + dq_{i_side}) < (\beta \gg 2)$, wherein \gg represents a right shift operator defined as

$$a \gg b = \left\lfloor \frac{a}{2^b} \right\rfloor \text{ and } \lfloor c \rfloor \text{ is a largest integer not greater than } c \text{ and } \beta \text{ is a parameter that depends}$$

on a quantization parameter value associated with said block (10) of pixels (12).

25

8. The method according to claim 6, further comprising calculating (S60) $| p_{0i} - q_{0i} |$, wherein selecting (S52) to apply strong deblocking filtering comprises selecting (S61) to apply strong deblocking filtering if $(dp_i + dq_i) < Thr1$, $(dp_{i_side} + dq_{i_side}) < Thr2$ and $| p_{0i} - q_{0i} | < Thr3$, wherein $Thr3$ denotes a third threshold value.

30

9. The method according to claim 8, wherein selecting (S61) to apply strong deblocking filtering comprises selecting (S61) to apply strong deblocking filtering if $2 \times (dp_i + dq_i) < (\beta \gg 2)$, $2 \times (dp_{i_side} + dq_{i_side}) < (\beta \gg 2)$ and $| p_{0i} - q_{0i} | < ((5 \times t_c + 1) \gg 1)$, wherein \gg represents a right

shift operator defined as $a \gg b = \left\lfloor \frac{a}{2^b} \right\rfloor$ and $\lfloor c \rfloor$ is a largest integer not greater than c and β and t_c are parameters that depend on a quantization parameter value associated with said block (10) of pixels (12).

- 5 10. The method according to claim 8, wherein selecting (S61) to apply strong deblocking filtering comprises selecting (S61) to apply strong deblocking filtering if $2 \times (dp_i + dq_i) < (\beta \gg 2)$, $2 \times (dp_{i_side} + dq_{i_side}) < (\beta \gg 2)$ and $|p0_i - q0_i| < (t_c \ll 4)$, wherein \gg represents a right shift operator defined as $a \gg b = \left\lfloor \frac{a}{2^b} \right\rfloor$ and $\lfloor c \rfloor$ is a largest integer not greater than c , \ll denotes a left shift operator defined as $a \ll b = a \times 2^b$ and β and t_c are parameters that depend on a quantization parameter value associated with said block (10) of pixels (12).

11. A filtering control device (100) comprising:

a determining unit (110) configured to determine whether pixels values of four pixels (12) in a line (15) of pixels (12, 22) in a block (10) of pixels (12) form an approximate line, and determine whether pixel values of four pixels (22) in said line (15) of pixels (12, 22) in a neighboring block (20) of pixels (22) form an approximate line; and

a processing unit (120) configured to select to apply strong deblocking filtering to pixel values in said line (15) of pixels (12, 22) in said block (10) of pixels (12) and said neighboring block (20) of pixels (22) if said pixel values of said four pixels (12) in said block (10) of pixels (12) form an approximate line and if said pixel values of said four pixels (22) in said neighboring block (20) of pixels (22) form an approximate line.

12. The filtering control device according to claim 11, wherein said processing unit (120) is configured to select to apply weak deblocking filtering to pixel values in said line (15) of pixels (12, 22) in said block (10) of pixels (12) and said neighboring block (20) of pixels (22) if said pixel values of said four pixels (12) in said block (10) of pixels (12) do not form an approximate line and/or if said pixel values of said four pixels (22) in said neighboring block (20) of pixels (22) do not form an approximate line.

- 30 13. The filtering control device according to claim 11 or 12, wherein

said determining unit (110) is configured to determine whether said pixels values of said four pixels (12) in said line (15) of pixels (12, 22) in said block (10) of pixels (12) form a flat line or a ramp, and determine whether said pixel values of said four pixels (22) in said line (15) of pixels (12, 22) in said neighboring block (20) of pixels (22) form a flat line or a ramp; and

5 said processing unit (120) is configured to select to apply strong deblocking filtering if said pixel values of said four pixels (12) in said block (10) of pixels (12) form a flat line or a ramp and if said pixel values of said four pixels (22) in said neighboring block (20) of pixels (22) form a flat line or a ramp.

14. The filtering control device according to any of the claims 11 to 13, wherein

10 said determining unit (110) is configured to determine whether a difference between i) a pixel value of a first pixel (12), relative to a block boundary (2) between said block (10) of pixels (12) and said neighboring block (20) of pixels (22), in said line (15) of pixels (12, 22) in said block (10) of pixels (12) and ii) a pixel value of a first pixel (22), relative to said block boundary (2), in said line (15) of pixels (12, 22) in said neighboring block (20) of pixels (22) is smaller than a threshold value; and

15 said processing unit (120) is configured to select to apply strong deblocking filtering if said pixel values of said four pixels (12) in said block (10) of pixels (12) form an approximate line, if said pixel values of said four pixels (22) in said neighboring block (20) of pixels (22) form an approximate line and if said difference is smaller than said threshold value.

20 15. The filtering control device according to any of the claims 11 to 14, wherein

 said determining unit (110) is configured to determine whether pixel values of a first pixel (12), a second pixel and a third pixel, relative to a block boundary (2) between said block (10) of pixels (12) and said neighboring block (20) of pixels (22), in said line (15) of pixels (12, 22) in said block (10) of pixels (12) form an approximate line, determine whether pixel values of said second pixel, said third
25 pixel and a fourth pixel, relative to said block boundary (2), in said line (15) of pixels (12, 22) in said block (10) of pixels (12) form an approximate line, determine whether pixel values of a first pixel (22), a second pixel and a third pixel, relative to said block boundary (2), in said line (15) of pixels (10, 12) in said neighboring block (20) of pixels (22) form an approximate line, and determine whether pixel values of said second pixel, said third pixel and a fourth pixel, relative to said block boundary (2), in said line
30 (15) of pixels (10, 12) in said neighboring block (20) of pixels (22) form an approximate line; and

 said processing unit (120) is configured to select to apply strong deblocking filtering if said pixel values of said first pixel (12), said second pixel and said third pixel in said block (10) of pixels (12) form an approximate line, if said pixel values of said second pixel, said third pixel and said fourth pixel in said block (10) of pixels (12) form an approximate line, if said pixel values of said first pixel (22), said

second pixel and said third pixel in said neighboring block (20) of pixels (22) form an approximate line, and if said pixel values of said second pixel, said third pixel and said fourth pixel in said neighboring block (20) of pixels (22) form an approximate line.

5 16. The filtering control device according to claim 15, wherein

said determining unit (110) comprises:

a dp_i calculator (111) configured to calculate $dp_i = | p_{2i} - 2 \times p_{1i} + p_{0i} |$, wherein p_{0i} denotes a pixel value of said first pixel (12) in said line (15) of pixels (12, 22) in said block (10) of pixels (12), p_{1i} denotes a pixel value of said second pixel in said line (15) of pixels (12, 22) in said block (10) of pixels (12), and p_{2i} denotes a pixel value of said third pixel in said line (15) of pixels (12, 22) in said
10 block (10) of pixels (12);

a dp_{i_side} calculator (113) configured to calculate $dp_{i_side} = | p_{3i} - 2 \times p_{2i} + p_{1i} |$, wherein p_{3i} denotes a pixel value of said third pixel in said line (15) of pixels (12, 22) in said block (10) of pixels (12);

15 a dq_i calculator (115) configured to calculate $dq_i = | q_{2i} - 2 \times q_{1i} + q_{0i} |$, wherein q_{0i} denotes a pixel value of said first pixel (22) in said line (15) of pixels (12, 22) in said neighboring block (20) of pixels (22), q_{1i} denotes a pixel value of said second pixel in said line (15) of pixels (12, 22) in said neighboring block (20) of pixels (22), and q_{2i} denotes a pixel value of said third pixel in said line (15) of pixels (12, 22) in said neighboring block (20) of pixels (22); and

20 a dq_{i_side} calculator (117) configured to calculate $dq_{i_side} = | q_{3i} - 2 \times q_{2i} + q_{1i} |$, wherein q_{3i} denotes a pixel value of said fourth pixel in said line(15) of pixels (12, 22) in said neighboring block (20) of pixels (22); and

said processing unit (120) is configured to select to apply strong deblocking filtering if $(dp_i + dq_i) < Thr1$ and $(dp_{i_side} + dq_{i_side}) < Thr2$, wherein $Thr1$ denotes a first threshold value and $Thr2$
25 denotes a second threshold value.

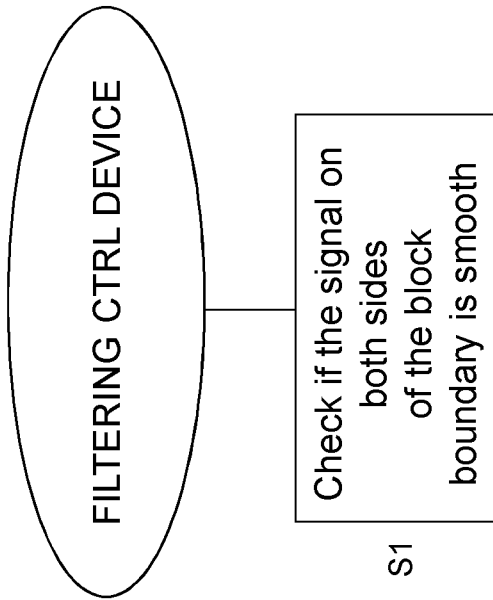
17. The filtering control device according to claim 16, wherein said processing unit (120) is configured to select to apply strong deblocking filtering if $2 \times (dp_i + dq_i) < (\beta \gg 2)$ and $2 \times (dp_{i_side}$

$+ dq_{i_side}) < (\beta \gg 2)$, wherein \gg represents a right shift operator defined as $a \gg b = \left\lfloor \frac{a}{2^b} \right\rfloor$ and

30 $\lfloor c \rfloor$ is a largest integer not greater than c and β is a parameter that depends on a quantization parameter value associated with said block (10) of pixels (12).

18. The filtering control device according to claim 16, wherein
said determining unit (110) comprises a $|p_{0i} - q_{0i}|$ calculator (119) configured to calculate $|p_{0i} - q_{0i}|$; and
said processing unit (120) is configured to select to apply strong deblocking filtering if $(dp_i + dq_i) < Thr1$, $(dp_{i_side} + dq_{i_side}) < Thr2$ and $|p_{0i} - q_{0i}| < Thr3$, wherein Thr3 denotes a third threshold value.
19. The filtering control device according to claim 18, wherein said processing unit (120) is configured to select to apply strong deblocking filtering if $2 \times (dp_i + dq_i) < (\beta \gg 2)$, $2 \times (dp_{i_side} + dq_{i_side}) < (\beta \gg 2)$ and $|p_{0i} - q_{0i}| < ((5 \times t_c + 1) \gg 1)$, wherein \gg represents a right shift operator defined as $a \gg b = \left\lfloor \frac{a}{2^b} \right\rfloor$ and $\lfloor c \rfloor$ is a largest integer not greater than c and β and t_c are parameters that depend on a quantization parameter value associated with said block (10) of pixels (12).
20. The filtering control device according to claim 18, wherein said processing unit (120) is configured to select to apply strong deblocking filtering if $2 \times (dp_i + dq_i) < (\beta \gg 2)$, $2 \times (dp_{i_side} + dq_{i_side}) < (\beta \gg 2)$ and $|p_{0i} - q_{0i}| < (t_c \ll 4)$, wherein \gg represents a right shift operator defined as $a \gg b = \left\lfloor \frac{a}{2^b} \right\rfloor$ and $\lfloor c \rfloor$ is a largest integer not greater than c , \ll denotes a left shift operator defined as $a \ll b = a \times 2^b$ and β and t_c are parameters that depend on a quantization parameter value associated with said block (10) of pixels (12).
21. An encoder (40) comprising a filtering control device (100) according to any of the claims 11 to 20.
22. A decoder (60) comprising a filtering control device (100) according to any of the claims 11 to 20.
23. A user equipment (80) comprising an encoder (40) according to claim 21 and/or a decoder (60) according to claim 22.

24. A network device (30) being or belonging to a network node in a communication network (32), said network device (30) comprising an encoder (40) according to claim 21 and/or a decoder (60) according to claim 23.
- 5 25. A computer (70) comprising:
a processing unit (72); and
a computer program product (73) comprising a computer program (74), said computer program (74) comprising code means (310), which when run on said processing unit (72), cause said processing unit (72) to:
- 10 a) check whether pixels values of four pixels (12) in a line (15) of pixels (12, 22) in a block (10) of pixels (12) form an approximate line;
b) check whether pixel values of four pixels (22) in said line (15) of pixels (12, 22) in a neighboring block (20) of pixels (22) form an approximate line; and
select to apply strong deblocking filtering to pixel values in said line (15) of pixels (12, 22)
15 in said block (10) of pixels (12) and said neighboring block (20) of pixels (22) if said pixel values of said four pixels (12) in said block (10) of pixels (12) form an approximate line and if said pixel values of said four pixels (22) in said neighboring block (20) of pixels (22) form an approximate line.
26. A computer program (74) for filtering control, said computer program (74) comprises code
20 means (310) which when executed by a processing unit (72) causes said processing unit (72) to:
a) check whether pixels values of four pixels (12) in a line (15) of pixels (12, 22) in a block (10) of pixels (12) form an approximate line;
b) check whether pixel values of four pixels (22) in said line (15) of pixels (12, 22) in a neighboring block (20) of pixels (22) form an approximate line; and
25 select to apply strong deblocking filtering to pixel values in said line (15) of pixels (12, 22) in said block (10) of pixels (12) and said neighboring block (20) of pixels (22) if said pixel values of said four pixels (12) in said block (10) of pixels (12) form an approximate line and if said pixel values of said four pixels (22) in said neighboring block (20) of pixels (22) form an approximate line.
- 30 27. A computer program product (73) comprising computer readable code means and a computer program (74) according to claim 26 stored on said computer readable code means.



S1

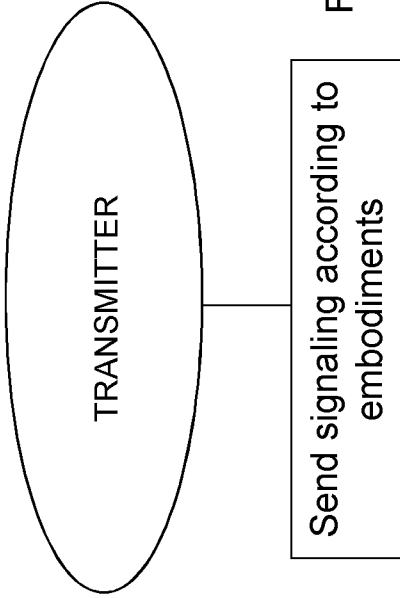


FIG. 1B

FIG. 1A

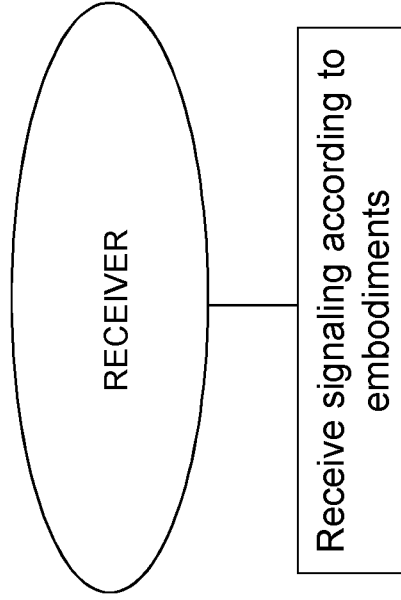


FIG. 1C

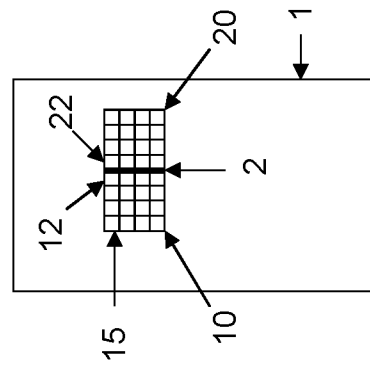


FIG. 17

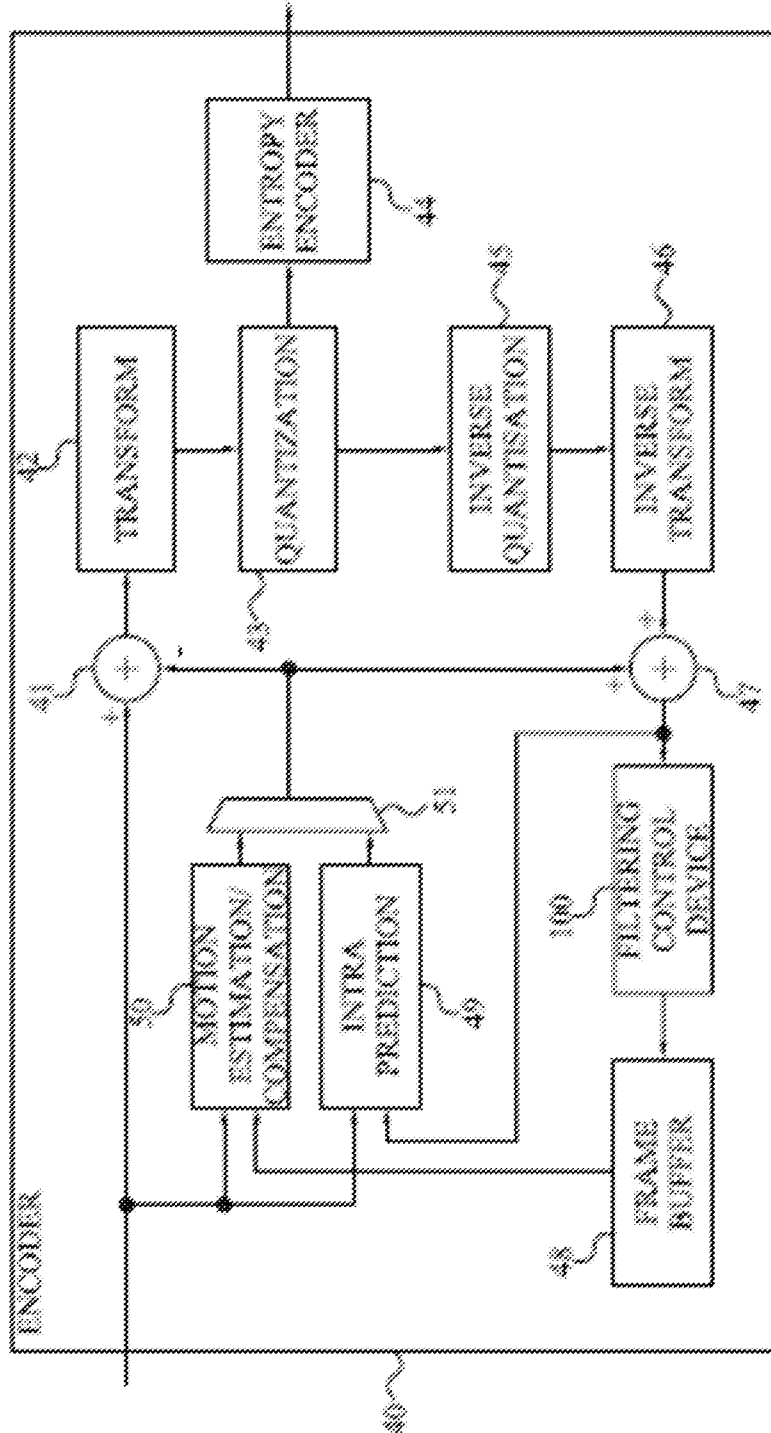


FIG. 2

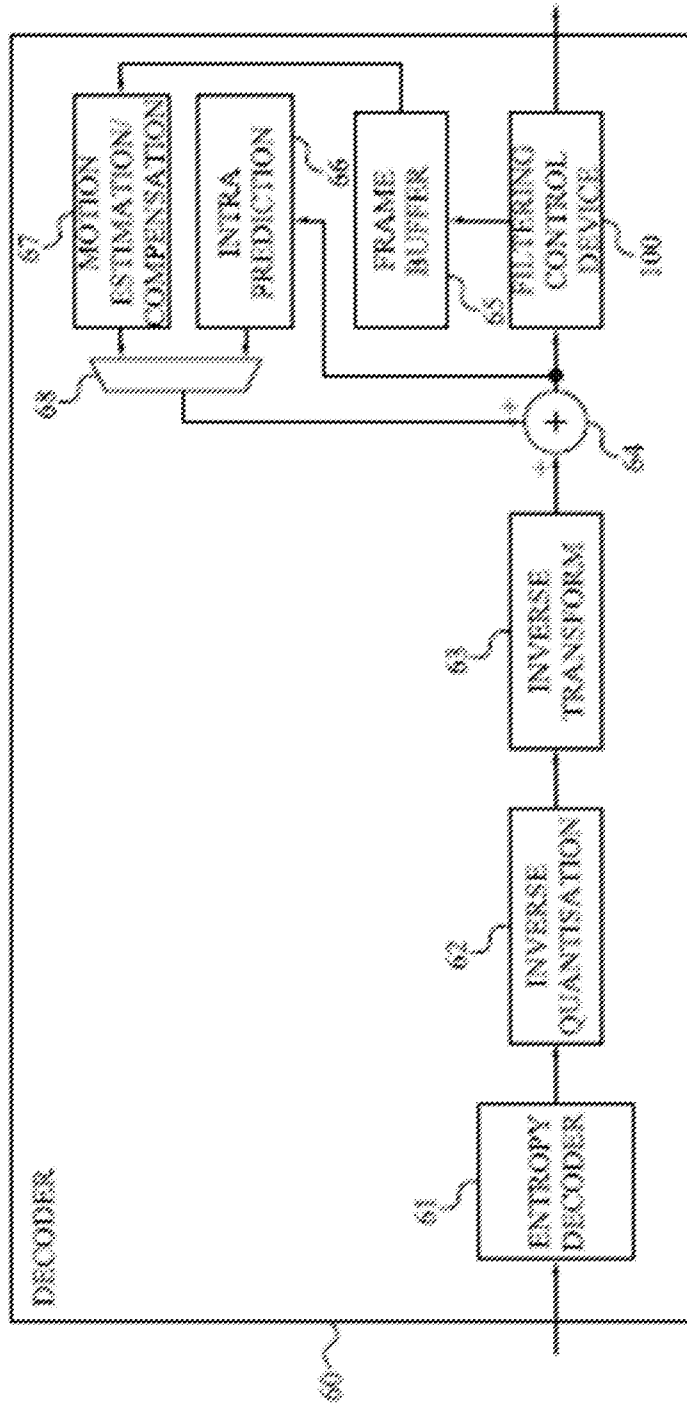


FIG. 3

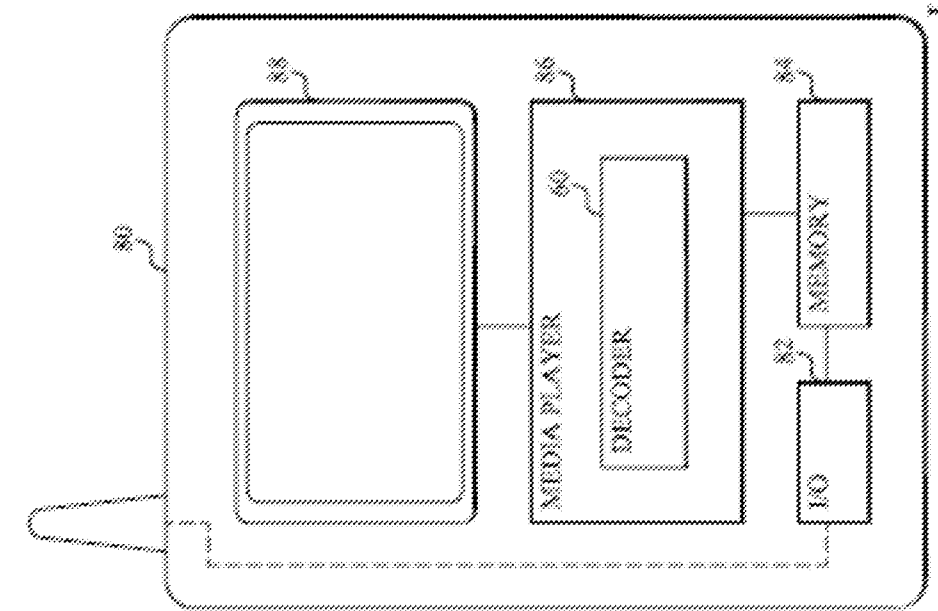


FIG. 5

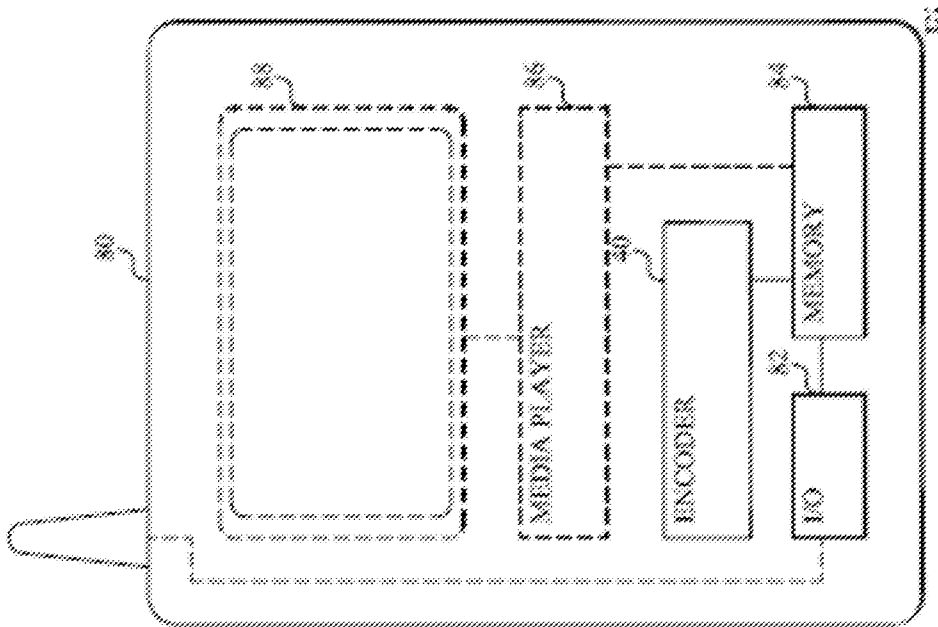


FIG. 4

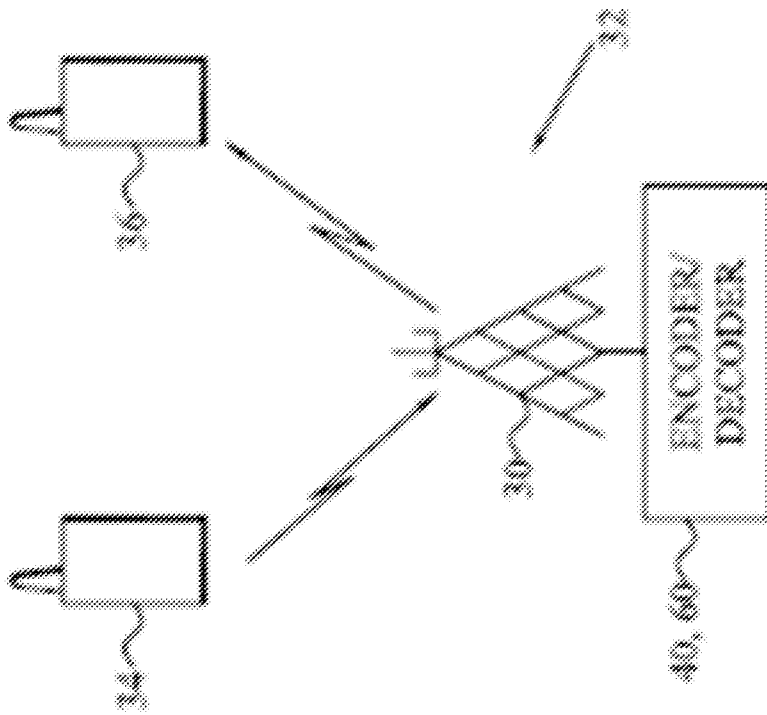
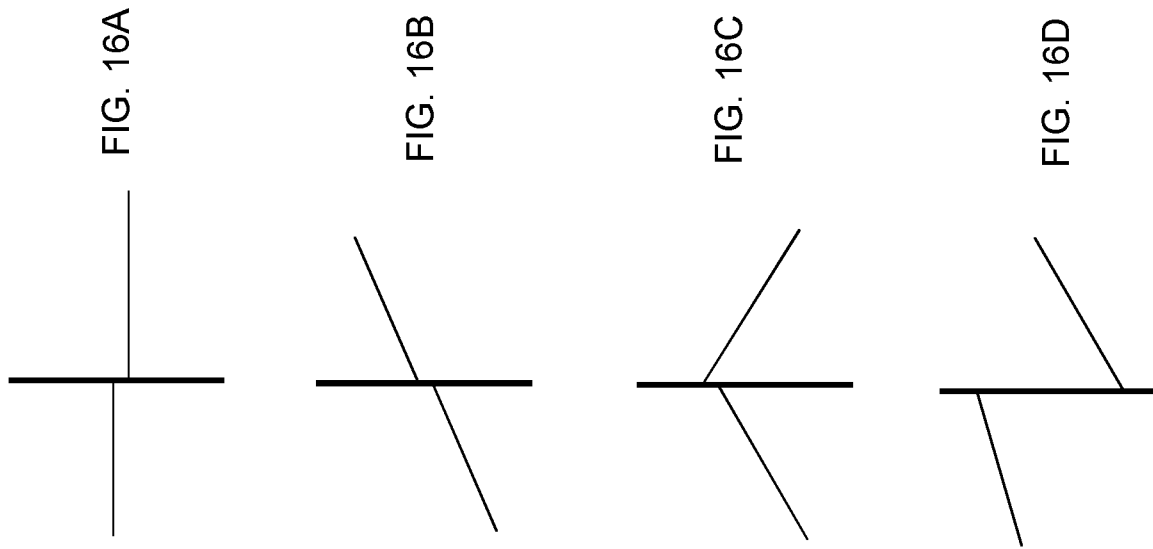


FIG. 6

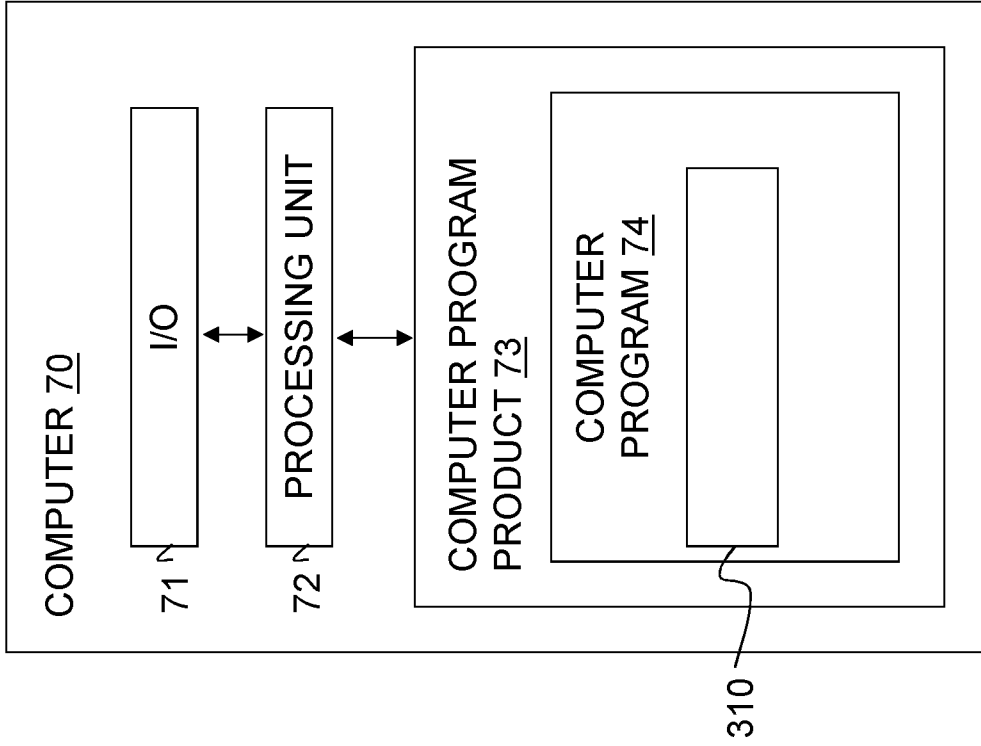


FIG. 8

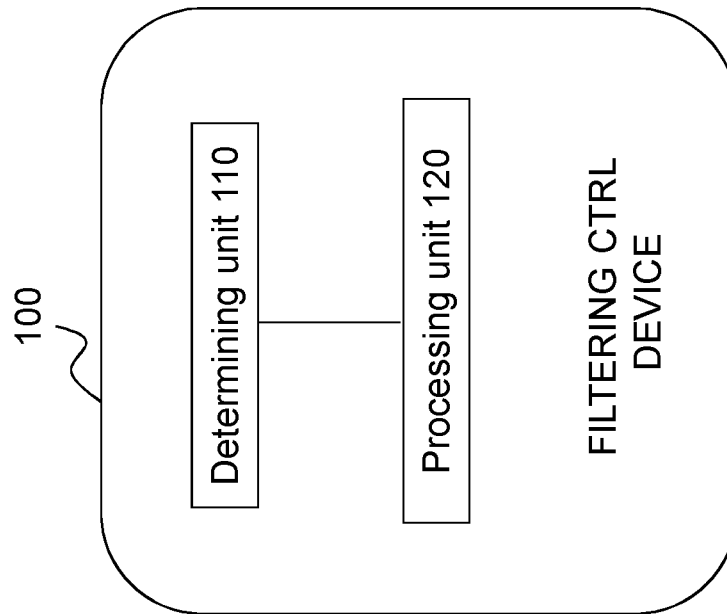
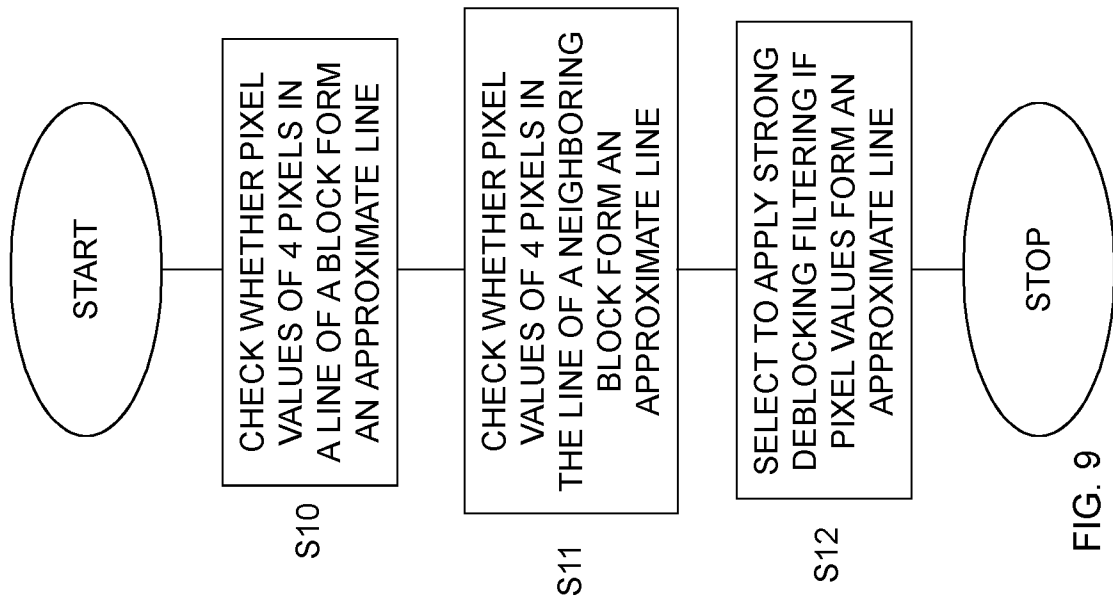
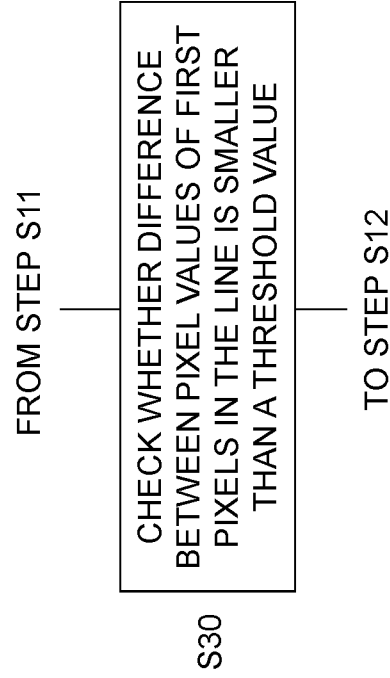
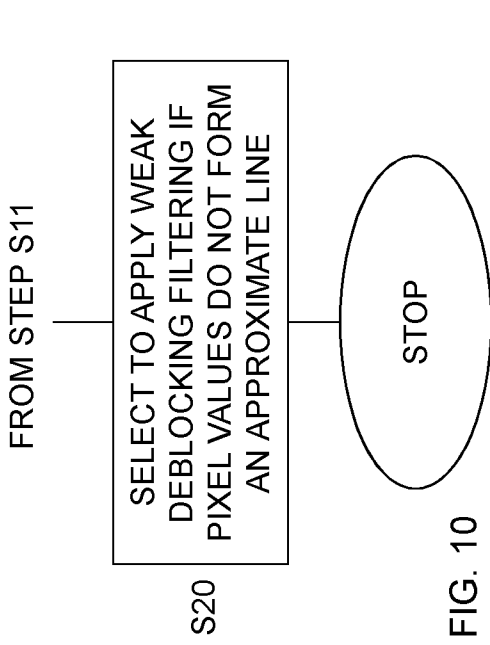


FIG. 7



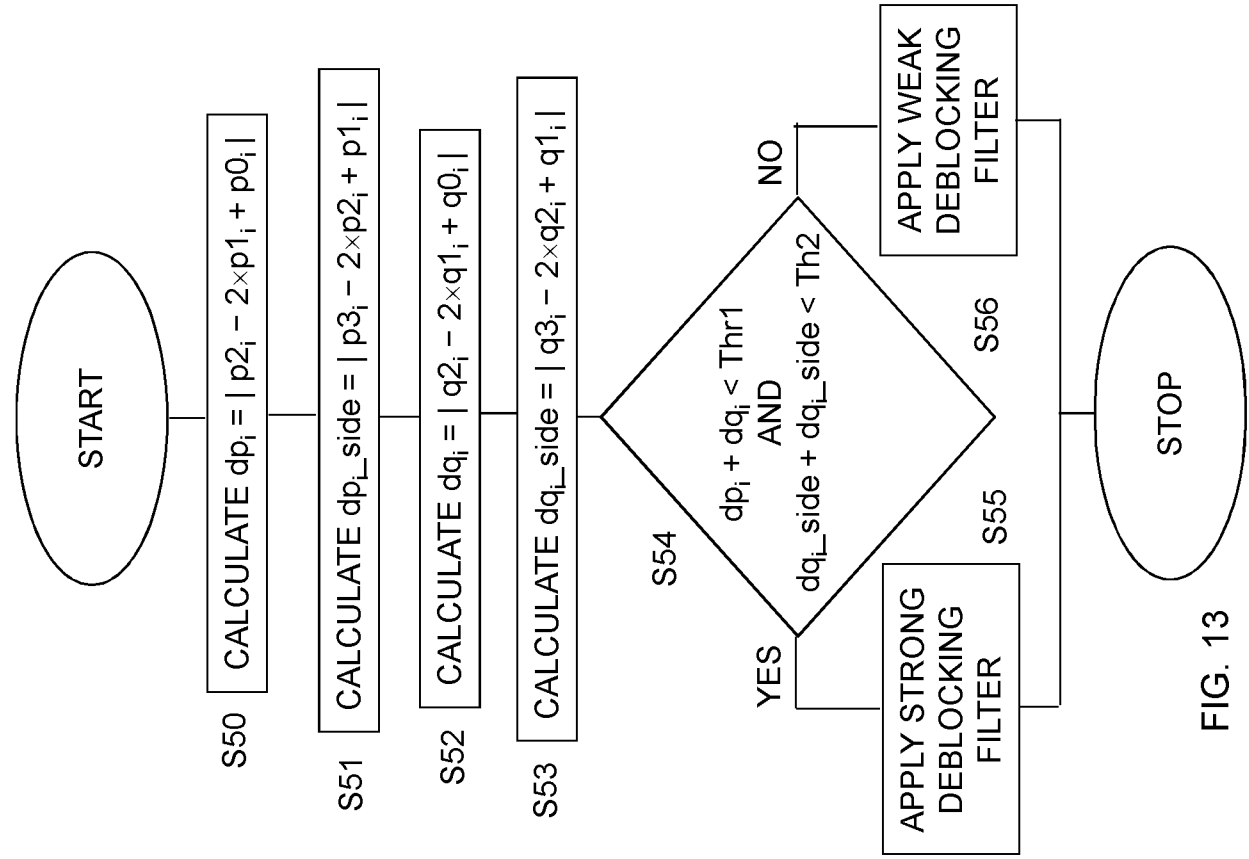


FIG. 13

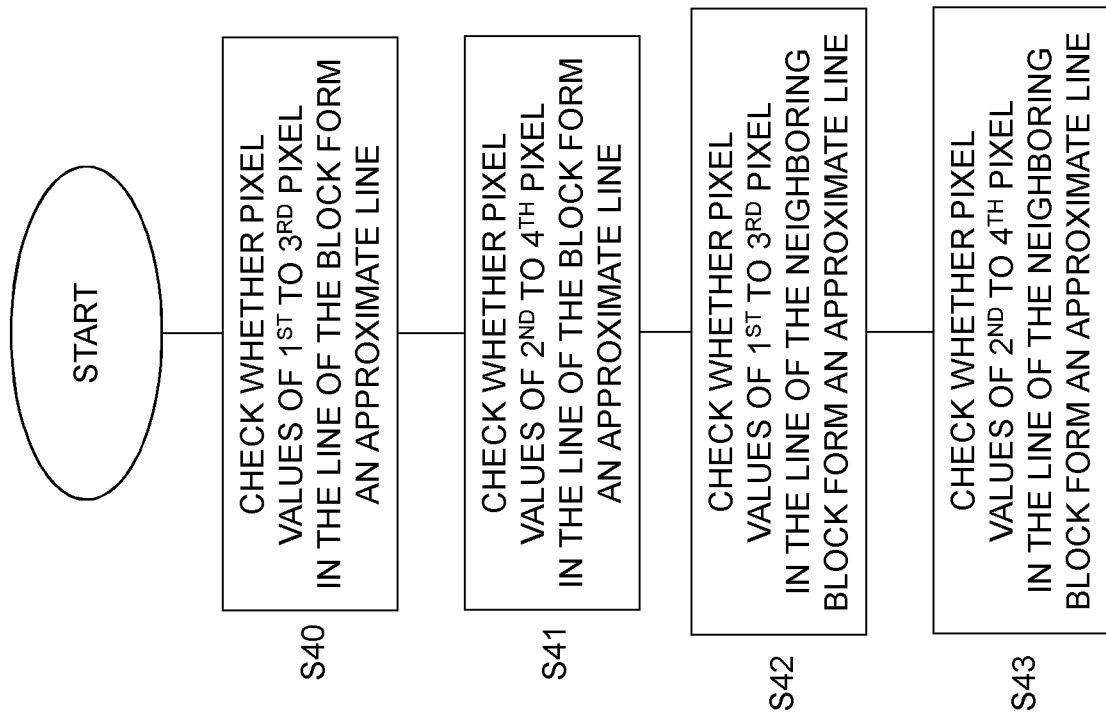


FIG. 12 TO STEP S12

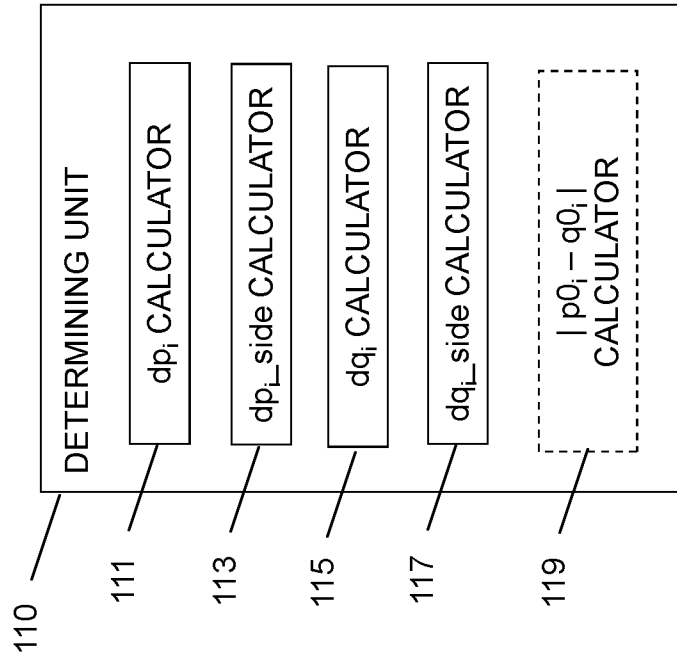


FIG. 15

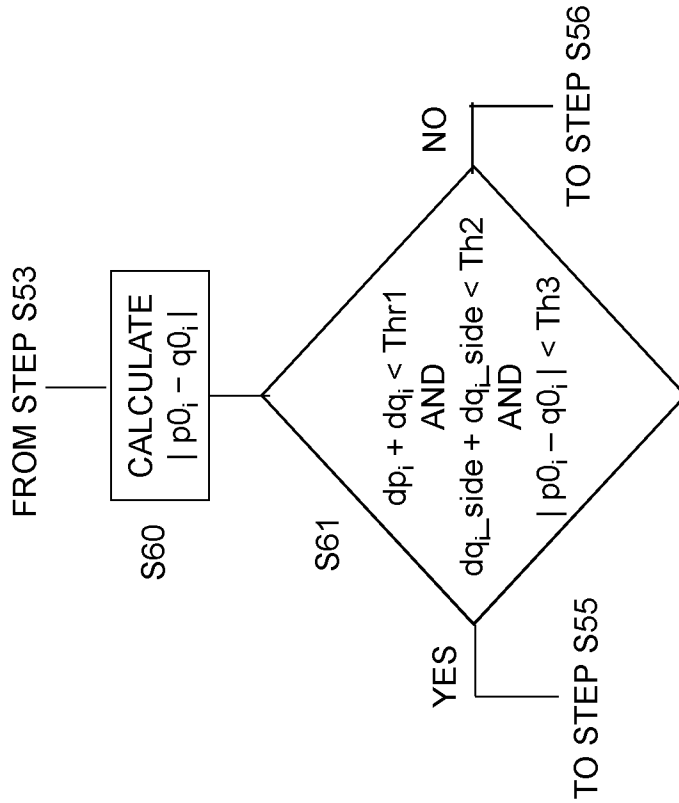


FIG. 14

INTERNATIONAL SEARCH REPORT

International application No PCT/SE2013/050833

A. CLASSIFICATION OF SUBJECT MATTER INV. H04N7/26 H04N7/50 ADD.				
According to International Patent Classification (IPC) or to both national classification and IPC				
B. FIELDS SEARCHED				
Minimum documentation searched (classification system followed by classification symbols) H04N				
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched				
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) EPO-Internal, WPI Data				
C. DOCUMENTS CONSIDERED TO BE RELEVANT				
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.		
X	US 2011/194614 A1 (NORKIN ANDREY [SE] ET AL) 11 August 2011 (2011-08-11) abstract page 3, line 6 - page 4, line 19 page 7, line 19 - page 15, line 19 page 20, line 16 - page 21, line 10 figures 1,4 -----	1-4, 11-14, 21-27		
X	WO 02/096117 A1 (PACE SOFT SILICON LTD [IE]; KARANDIKAR PRAVIN [IN]; SATYANARAYANA UPPA) 28 November 2002 (2002-11-28) page 6, line 1 - page 9, line 26; figures 1,2 ----- -/--	1,11, 21-27		
<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C. <input checked="" type="checkbox"/> See patent family annex.				
* Special categories of cited documents : <table style="width: 100%; border: none;"> <tr> <td style="width: 50%; border: none; vertical-align: top;"> "A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier application or patent but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed </td> <td style="width: 50%; border: none; vertical-align: top;"> "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family </td> </tr> </table>			"A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier application or patent but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family
"A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier application or patent but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family			
Date of the actual completion of the international search	Date of mailing of the international search report			
24 October 2013	04/11/2013			
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Authorized officer Le Guen, Benjamin			

INTERNATIONAL SEARCH REPORT

International application No PCT/SE2013/050833

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	<p>ANDREY NORKIN ET AL: "CE12.1: Ericsson deblocking filter", 96. MPEG MEETING; 21-3-2011 - 25-3-2011; GENEVA; (MOTION PICTURE EXPERT GROUP OR ISO/IEC JTC1/SC29/WG11),, no. m19803, 19 March 2011 (2011-03-19), XP030048370, paragraph [02.1] - paragraph [2.2.2] -----</p>	1-27

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/SE2013/050833

Patent document cited in search report	Publication date	Patent family member(s)	Publication date	
US 2011194614	A1	11-08-2011	AU 2010345142 A1	23-08-2012
			CA 2785036 A1	11-08-2011
			CA 2803508 A1	11-08-2011
			CN 102860005 A	02-01-2013
			EP 2532161 A1	12-12-2012
			EP 2564594 A1	06-03-2013
			JP 2013519293 A	23-05-2013
			US 2011194614 A1	11-08-2011
			US 2013051480 A1	28-02-2013
			WO 2011096869 A1	11-08-2011
			WO 2011096876 A1	11-08-2011

WO 02096117	A1	28-11-2002	NONE	
