CLAIMS:

1. A method (400) for deblocking filtering of vertical and/or horizontal block boundaries, the method being performed by an image or video encoder or an image or video decoder, the method comprising:

obtaining (s402) a first set of sample values comprising a first block of sample values and a second block of sample values, wherein the first block of sample values comprises a set of sample values B1_i,k for i=0-7 and k=0-3, and the second block of sample values comprises a set of sample values B2_j,k for j=0-7 and k=0-3; and

deciding (s404) whether or not to use a long filter to filter at least a subset of the first set of sample values, wherein deciding whether or not to use a long filter to filter at least a subset of the first set of sample values comprises:

calculating (s502) a first gradient value, maxGradB1_6,0, using sample value B1_6,0;

calculating (s504) a first pq value, pq1_0, using maxGradB1_6,0, wherein calculating pq1_0 comprises calculating a value sp, calculating a value sq, and calculating sp + sq, wherein pq1_0 = (sp + sq), and wherein calculating sp comprises calculating sp = (Abs(B1_0,0 – B1_3,0) + Abs(B1_3,0 – B1_7,0) + maxGradB1_6,0 + 1) >> 1 and wherein maxGradB1_6,0 equals Abs(B1_7,0 - B1_6,0 - B1_5,0 + B1_4,0);

comparing (s508) pq1_0 to a threshold;

calculating a second gradient value, maxGradB1_6,3, using sample value B1_6,3;

calculating a second pq value, pq1_3, using maxGradB1_6,3; and

comparing (s508) pq1_3 to the threshold.

2. The method of claim 1, wherein

the second block has a maximum filter length, maxFilterLengthQ,

if maxFilterLengthQ is equal to 7, then sq = (Abs(B2_0,0 – B2_3,0) + Abs(B2_3,0 – B2_7,0) + maxGradB2_6,0 + 1) >> 1, where maxGradB2_6,0 is a third gradient value,

if maxFilterLengthQ is equal to 5, then sq = (Abs(B2_0,0 – B2_3,0) + Abs(B2_3,0 – B2_5,0) + 1) >> 1, and

if maxFilterLengthQ is not equal to 7 and is not equal to 5, then sq = Abs(B2_0,0 – B2_3,0).

3. The method of claim 2, wherein maxGradB2_6,0 equals Abs(B2_7,0 – B2_6,0 – B2_5,0 + B2_4,0).

4. The method of any one of claims 1-3, wherein calculating pq1_3 comprises calculating a value sp3, calculating a value sq3, and calculating sp3 + sq3, wherein pq1_3 = (sp3 + sq3), and wherein calculating sp3 comprises calculating sp3 = (Abs(B1_0,3 – B1_3,3) + Abs(B1_3,3 – B1_7,3) + maxGradB1_6,3 + 1) >> 1.

5. The method of claim 4, wherein maxGradB1_6,3 equals Abs(B1_7,3 - B1_6,3 - B1_5,3 + B1_4,3).

6. The method of any one of claims 1-5, wherein

the step of calculating pq1_0 using maxGradB1_6,0 is performed as a result of determining that a maximum filter length for the first block, maxFilterLengthP, is equal to 7.

7. The method of any one of claims 1-6, wherein

the step of calculating pq1_3 using maxGradB1_6,3 is performed as a result of determining that a max filter length for the first block, maxFilterLengthP, is equal to 7.

8. The method of any one of claims 1-7, wherein

when deblocking a vertical boundary, the i value identifies a column of a block and the k value identifies a row of the block, and

when deblocking a horizontal boundary, the k value identifies a column of the block and the i value identifies a row of the block.

9. The method of any one of claims 1-8, wherein the samples to filter by the long filter when maxFilterLengthP is equal to 7 are B1_0,0 to B1_6,0, B1_0,1 to B1_6,1, B1_0,2 to B1_6,2 and B1_0,3 to B1_6,3.

10. The method of any one of claims 1-9, wherein the samples to filter by the long filter

when maxFilterLengthQ is equal to 7 are B2_0,0 to B2_6,0, B2_0,1 to B2_6,1, B2_0,2 to B2_6,2 and B2_0,3 to B2_6,3.

11. A method (400) for deblocking filtering of vertical and/or horizontal block boundaries, the method being performed by an image or video encoder or an image or video decoder, the method comprising:

obtaining (s402) a first set of sample values comprising a first block of sample values and a second block of sample values, wherein the first block of sample values comprises a set of sample values B1_i,k for i=0-7 and k=0-3, and the second block of sample values comprises a set of sample values B2_j,k for j=0-7 and k=0-3; and

selecting (s404) a long filter, wherein selecting the long filter comprises:

calculating (s504) a first pq value, pq2_0, using at least a first value equal to Abs(B1_2,0 - 2 * B1_1,0 + B1_0,0) and a second value equal to Abs(B2_2,0 - 2 * B2_1,0 + B2_0,0); and

comparing (s508) pq2_0 to a threshold, wherein

the threshold is equal to $\beta >> 4$, and

β is selected from a set of predetermined beta values based on a quantization parameter value, Q.

12. The method of claim 11, wherein selecting the long filter further comprises calculating a second pq value, pq2_3, using at least a first value equal to Abs(B1_2,3 - 2 * B1_1,3 + B1_0,3) and a second value equal to Abs(B2_2,3 - 2 * B2_1,3 + B2_0,3); and

comparing pq2_3 to the threshold.

13. A method for deblocking filtering of vertical and/or horizontal block boundaries, the method being performed by an image or video encoder or an image or video decoder, the method comprising:

1) obtaining (s402) a first set of sample values comprising a first block of sample values and a second block of sample values, wherein the first block of sample values comprises a set of sample values B1_i,k for i=0-7 and k=0-3, and the second block of sample values comprises a set of sample values B2_j,k for j=0-7 and k=0-3; and

2) selecting (s404) a long filter, wherein selecting the long filter comprises:

i) calculating a first pq value, pq1_0, wherein calculating pq1_0 comprises calculating a value sp, calculating a value sq, and calculating sp + sq, wherein pq1_0 = (sp + sq);

ii) comparing pq1_0 to a first threshold;

iii) calculating (s504) a second pq value, pq2_0, using at least a first value equal to Abs(B1_2,0 - 2 * B1_1,0 + B1_0,0) and a second value equal to Abs(B2_2,0 - 2 * B2_1,0 + B2_0,0); and

iv) comparing (s508) pq2_0 to a second threshold, wherein the second threshold is equal to $\beta >> 4$, and β is selected from a set of predetermined beta values based on a quantization parameter value, Q.

14. The method as in claim 13 wherein selecting the long filter further comprises:

calculating a third pq value, pq3_0, wherein pq3_0 = Abs(B1_0,0 - B2_0,0); and

comparing pq3_0 with a third threshold, wherein

the third threshold is equal to (5*tC+1) >> 1, wherein tC is selected from a set of predetermined values based on a quantization parameter value, Q.

15. The method as in claim 13 or 14, wherein selecting the long filter further comprises:

calculating a fourth pq value, pq2_3;

comparing pq2_3 to the second threshold;

calculating a fifth pq value, pq1_3; and

comparing pq1_3 to the first threshold, wherein

the step of calculating pq2_3 comprises calculating pq2_3 using at least a first value equal to Abs(B1_2,3 - 2 * B1_1,3 + B1_0,3) and a second value equal to Abs(B2_2,3 - 2 * B2_1,3 + B2_0,3), and when maxFilterLengthP is greater than 3 also using a third value Abs(B1_5,3 – 2*B1_4,3 + B1_3,3) and when maxFilterLengthQ is greater than 3 also using a fourth value Abs(B2_5,3 – 2*B2_4,3 + B2_3,3),

the step of calculating pq1_3 comprises: calculating a value sp3, calculating a value sq3, and calculating sp3 + sq3, where pq1_3 = (sp3 + sq3).

16. The method of any one of claims 13-15, wherein

the first block has a maximum filter length, maxFilterLengthP,

if maxFilterLengthP is equal to 7, then calculating sp comprises calculating sp = (Abs(B1_0,0 – B1_3,0) + Abs(B1_3,0 – B1_7,0) + maxGradB1_6,0 + 1) >> 1, where maxGradB1_6,0 is a gradient value,

if maxFilterLengthP is 5, then calculating sp comprises calculating sp=(Abs(B1_0,0 – B1_3,0) + Abs(B1_3,0 – B1_5,0) + 1) >> 1, and

if maxFilterLengthP is equal to 3, then calculating sp comprises calculating sp=Abs(B1_0,0 – B1_3,0).

17. The method of any one of claims 13-16, wherein

the second block has a maximum filter length, maxFilterLengthQ,

if maxFilterLengthQ is equal to 7, then calculating sq comprises calculating sq = (Abs(B2_0,0 – B2_3,0) + Abs(B2_3,0 – B2_7,0) + maxGradB2_6,0 + 1) >> 1, where maxGradB2_6,0 is a gradient value,

if maxFilterLengthQ is equal to 5, then calculating sq comprises calculating sq=(Abs(B2_0,0 – B2_3,0) + Abs(B2_3,0 – B2_5,0) + 1) >> 1, and

if maxFilterLengthQ is equal to 3, then calculating sq comprises calculating sq= Abs(B2_0,0 – B2_3,0).

18. A computer program (643) comprising instructions (644) which when executed by processing circuitry (602) causes the processing circuitry (602) to perform the method of any one of claims 1-17.

19. A carrier containing the computer program of claim 18, wherein the carrier is one of an electronic signal, an optical signal, a radio signal, and a computer readable storage medium (642).

20. An apparatus (600) for deblocking filtering of vertical and/or horizontal block

boundaries, the apparatus being adapted to perform the method according to any of claims 1-17.