

CLAIMS

1. A method for processing a reconstructed block, wherein the reconstructed block
5 comprises a plurality of pixels, wherein the method comprises:
 scanning a current pixel of the reconstructed block and its neighboring pixels of the
current pixel according to a predefined scan template;
 obtaining spectrum components by performing transform for the current pixel and its
neighboring pixels;
10 determining whether a filter is applied to the spectrum components based on
conditions;
 when the filter is applied to spectrum component, obtaining filtered spectrum
components based on a filtering parameter and the spectrum components;
 when the filter is not applied to spectrum component, saving spectrum component
15 without changing;
 obtaining filtered pixels by performing inverse transform for the filtered spectrum
components; and
 generating a filtered reconstructed block based on the filtered pixels.
- 20 2. The method of claim 1, wherein the conditions comprises at least one of:
 the block has non-zero residual signal;
 a size of the block;
 an aspect ratio of the block;
25 a prediction mode of the block;
 a prediction mode of the block in intra; or
 a prediction mode of the block in inter.
3. The method of claim 1 or 2, wherein the filter parameter and/or the predefined scan
30 template are depending on the conditions.
4. The method of claim 2, wherein the filter is applied if size of the block obtained as block
width multiplied by block height is greater than 4x4.
- 35 5. The method of claim 2, wherein the filter is applied if the prediction mode of the block is
inter and the block has non-zero residual signal.

6. The method of claim 2, wherein the filter is applied if the prediction mode of the block is intra.

7. The method of claim 2, wherein the filter is not applied if the prediction mode of the block is inter and the block width and the block height is greater or equal to 32.

8. The method of claim 7, wherein the obtaining the filtered spectrum components are estimated based on a look up table (LUT).

9. The method of claim 8, a filtered spectrum component $F(i, \sigma)$ is derived by:

$$F(i, \sigma) = \begin{cases} R(i) & , \text{Abs}(R(i)) \geq THR \\ LUT(R(i), \sigma), & R(i) > 0 \\ -LUT(-R(i), \sigma), & R(i) \leq 0 \end{cases}$$

wherein (i) is an index of a spectrum component, $R(i)$ is the spectrum component

corresponding to (i) index, σ is the filtering parameter, $LUT(R(i), \sigma)$ is an array in the look up table, and THR is a threshold.

10. The method of claim 9, wherein $LUT(R(i), \sigma) = \frac{R(i)^3}{R(i)^2 + m \cdot \sigma^2}$, m is normalization constant equal to number of spectrum components.

11. The method of any one of claims 8-10, wherein the filtering parameter is derived based on a codec quantization parameter (QP).

12. The method of claim 11, wherein the filtering parameter σ is derived by

$$\sigma = k * 2^{(n \cdot (QP - s))}$$

, wherein QP is the codec quantization parameter, k, n and s are constants.

13. The method of claim 12, wherein k is modified based on bit depth of pixels $k_{mod} = k * (1 \ll (\text{bit_depth} - 8))$.

14. The method of claim 13, wherein the filtering parameter is equal to zero if $QP \leq 17$, and the reconstructed block is bypassed without filtering.

15. The method of any one of claims 9, THR is dependent of QP value.

16. The method of claim 15, THR is derived by:

$$\frac{THR^2}{THR^2 + m \cdot \sigma^2} = C$$

Wherein C is a value close to 1.

- 5 17. The method of any one of claims 1-16, wherein the predefined scan template is defined as set of spatial or raster offsets relative to a position of the current pixel inside the reconstructed block.
- 10 18. The method of claim 17, wherein when the predefined scan template is defined as set of raster offsets relative to the position of the current pixel inside the reconstructed block, offsets point to neighbour pixels are inside the reconstructed block.
- 15 19. The method of claim 18, wherein the predefined scan template is defined as follows:
 scanTpl[4] (y, x) = { (0,0), (0,1), (1,0), (1,1) }
- 20 20. The method of any one of claims 1-19, wherein at least one filtered pixel is placed to its original position according to the predefined scan template.
21. The method of any one of claims 1-20, wherein all filtered pixels are added to an accumulation buffer according to the predefined scan template, and wherein the accumulation buffer is initialized by zero before the obtaining filtered spectrum components.
- 25 22. The method of claim 21, wherein final filtered pixels are obtained as accumulated values in the accumulation buffer divided by number of pixels adding to a current position of the accumulation buffer; and
 wherein generating the filtered reconstructed block based on the final filtered pixels.
- 30 23. The method of any one of claims 1-22, wherein differences between all filtered and corresponding unfiltered pixels are added to an accumulation buffer according to the predefined scan template, and wherein the accumulation buffer is initialized by unfiltered pixels multiplied by maximum number of pixel values to be added in the block.
- 35 24. The method of claim 23, wherein final filtered pixels are obtained as accumulated values in the accumulation buffer divided by maximum number of pixel values to be added in the block.

25. The method of any one of claims 1-24, scanning the current pixel of the reconstructed block and its neighboring pixels of the current pixel according to the predefined scan template to an accumulation buffer; and
- 5 obtaining spectrum components by performing transform for pixels in the accumulation buffer.
26. The method of any one of claims 1-25, wherein the transform is a 1D transform
27. The method of any one of claims 1-26, wherein the transform is a Hadamard transform.
- 10 28. A decoder comprising processing circuitry for carrying out the method according to any one of claims 1-27.
29. An encoder comprising processing circuitry for carrying out the method according to any one of claims 1-27.
- 15 30. A decoder comprising a filter for carrying out the method according to any one of claims 1-27.
- 20 31. An encoder comprising a filter for carrying out the method according to any one of claims 1-27.
32. A decoder, comprising:
a memory storage comprising instructions; and
25 one or more processors in communication with the memory, wherein the one or more processors execute the instructions to carry out the method according to any one of claims 1-27.
33. An encoder, comprising:
30 a memory storage comprising instructions; and
one or more processors in communication with the memory, wherein the one or more processors execute the instructions to carry out the method according to any one of claims 1-27.
- 35 34. A computer program product comprising program code for performing the method of claims 1-27 when executed on a computer or a processor.