

CLAIMS

1. A method for processing a reconstructed block, wherein the reconstructed block 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;
- obtaining filtered spectrum components based on a filtering parameter and the spectrum components;
- obtaining filtered pixels by performing inverse transform for the filtered spectrum components; and
- generating a filtered reconstructed block based on the filtered pixels.

2. The method of claim 1, wherein the obtaining the filtered spectrum components by using a look up table (LUT).

3. The method of claim 2, 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.

4. The method of claim 3, 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.

5. The method of any one of claims 2-4, wherein the filtering parameter is derived based on a codec quantization parameter (QP).

6. The method of claim 5, wherein the filtering parameter σ is derived by

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

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

7. The method of claim 6, wherein $k = 2.64$, $n = 0.1296$ and $s = 11$.

8. The method of claim 6 or 7, wherein k is modified based on bit depth of pixels
 $k_{mod} = k * (1 < (bit_depth - 8))$.

9. The method of claim 5, wherein the filtering parameter σ is equal to zero if $QP \leq 17$, and the reconstructed block is bypassed without filtering.

10. The method of any one of claims 2-9, THR is dependent of QP value.

11. The method of claim 10, THR is derived by:

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

Wherein C is a value close to 1.

12. The method of any one of claims 1-11, 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.

13. The method of claim 12, wherein the predefined scan template is defined as follows: $scanTmpl[4](y, x) = \{ (0,0), (0,1), (1,0), (1,1) \}$

14. The method of claim 12, 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. The method of any one of claims 1-14, wherein at least one filtered pixel is placed to its original position according to the predefined scan template.

16. The method of any one of claims 1-15, 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.

17. The method of claim 16, 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 the generating the filtered reconstructed block based on the filtered pixels comprises: generating the filtered reconstructed block based on the final filtered pixels.

18. The method of any one of claims 1-17, 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.

19. The method of claim 18, 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.

20. The method of any one of claims 1-19, wherein the method is applied depending on conditions, 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;
- a prediction mode of the block;
- a prediction mode of the block in intra; or
- a prediction mode of the block in inter.

21. The method of claim 20, wherein the filter parameter and/or the predefined scan template are depending on the conditions.

22. The method of claim 20, where the method is applied if size of the block obtained as block width multiplied by block height is greater than 4x4.

23. The method of claim 20, where the method 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.

24. The method of claim 20, where the method is applied if the prediction mode of the block is inter and the block has non-zero residual signal.

25. The method of claim 20, where the method is applied if the prediction mode of the block is intra.

26. The method of any one of claims 1-25, 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
obtaining spectrum components by performing transform for pixels in the accumulation buffer.

27. The method of any one of claims 1-26, wherein the transform is a Hadamard transform.

28. The method of any one of claims 1-27, wherein the transform is a 1D transform.

29. A method for processing a predicted block, wherein the predicted block comprises a plurality of pixels, wherein the method comprises:

scanning a current pixel of the predicted 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;

obtaining filtered spectrum components based on a filtering parameter and the spectrum components;

obtaining filtered pixels by performing inverse transform for the filtered spectrum components; and

generating a filtered predicted block based on the filtered pixels.

30. The method of claim 29, wherein the obtaining the filtered spectrum components by using a look up table (LUT).

31. The method of claim 30, 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.

32. The method of claim 31, 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.

33. The method of any one of claims 30-32, wherein the filtering parameter is derived based on a codec quantization parameter (QP).

34. The method of claim 33, wherein the filtering parameter σ is derived by

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

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

35. The method of claim 33, wherein the filtering parameter σ is equal to zero if $QP \leq 17$, and the predicted block is bypassed without filtering.

36. The method of any one of claims 31-35, THR is dependent of QP value.

37. The method of claim 36, THR is derived by:

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

Wherein C is a value close to 1.

38. The method of any one of claims 29-37, wherein the predefined scan template is defined as set of spatial or raster offsets relative to a position of the current pixel inside the predicted block.

39. The method of claim 38, wherein the predefined scan template is defined as follows: $\text{scanTmp}[4](y, x) = \{ (0,0), (0,1), (1,0), (1,1) \}$
40. The method of any one of claims 29-39, wherein at least one filtered pixel is placed to its original position according to the predefined scan template.
41. The method of any one of claims 29-40, 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.
42. The method of claim 41, 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 the generating the filtered predicted block based on the filtered pixels comprises: generating the filtered predicted block based on the final filtered pixels.
43. The method of any one of claims 29-42, wherein the method is applied depending on conditions, 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;
a prediction mode of the block;
a prediction mode of the block in intra; or
a prediction mode of the block in inter.
44. The method of claim 43, wherein the filter parameter and/or the predefined scan template are depending on the conditions.
45. The method of claim 43, where the method is applied if size of the block obtained as block width multiplied by block height is greater than 4x4, or if the prediction mode of the block is intra, or if the prediction mode of the block is inter and the block has non-zero residual signal.
46. The method of any one of claims 29-45, wherein the transform is a Hadamard

transform.

47. A decoder comprising processing circuitry for carrying out the method according to any one of claims 1-46.

48. An encoder comprising processing circuitry for carrying out the method according to any one of claims 1-46.

49. A decoder comprising a filter for carrying out the method according to any one of claims 1-46.

50. An encoder comprising a filter for carrying out the method according to any one of claims 1-46.

51. A decoder, comprising:
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 1-46.

52. An encoder, comprising:
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 1-46.

53. A computer program product comprising program code for performing the method of claims 1-46 when executed on a computer or a processor.