

WHAT IS CLAIMED IS:

1. A method of video processing, comprising:
using a dimension of a virtual pipeline data unit (VPDU) used for a conversion between a video comprising one or more video regions comprising one or more video blocks and a bitstream representation of the video to perform a determination of whether a ternary-tree (TT) or a binary tree (BT) partitioning of a video block of the one or more video blocks is enabled;
and
performing, based on the determination, the conversion,
wherein the dimension is equal to VSize in luma samples, wherein dimensions of the video block are CtbSizeY in luma samples, wherein $VSize = \min(M, CtbSizeY)$, and wherein M is a positive integer.
2. The method of claim 1, wherein $M = 64$.
3. The method of claim 1 or 2, wherein the dimension of the VPDU is a height or a width.
4. The method of claim 1, wherein the determination is independent of a maximum transform size.
5. The method of any of claims 1 to 4, wherein VSize is a predetermined value.
6. The method of claim 5, wherein $VSize = 64$.
7. The method of claim 1, wherein determination of the TT partitioning is based on a width or a height of the video block in luma samples being greater than $\min(VSize, \maxTtSize)$, and wherein $\maxTtSize$ is a maximum transform size.
8. The method of claim 1, wherein determination of the TT partitioning is based on a width or a height of the video block in luma samples being greater than VSize.
9. The method of claim 7, wherein $\maxTtSize$ is 32×32 and VSize is 64×64 , and wherein the TT partitioning is disabled when a size of the video block is 128×128 , 128×64 , or 64×128 .
10. The method of claim 7, wherein $\maxTtSize$ is 32×32 and VSize is 64×64 , and wherein the TT partitioning is enabled when a size of the video block is 64×64 .

11. The method of claim 1, wherein the determination of a vertical BT partitioning is based on a width of the video block in luma samples being less than or equal to VSize and a height of the video block in luma samples being greater than VSize.
12. The method of claim 11, wherein a maximum transform size is 32×32 and VSize is 64×64 , and wherein the vertical BT partitioning is disabled when a size of the video block is 64×128 .
13. The method of claim 11, wherein a maximum transform size is 32×32 and VSize is 64×64 , and wherein the vertical BT partitioning is enabled when a size of the video block is 32×64 , 16×64 , or 8×64 .
14. The method of claim 1, wherein a vertical BT partitioning is disabled when (i) a sum of a width of the video block in luma samples and a horizontal coordinate of a top-left luma sample of the video block is greater than a width of a picture or a width of a subpicture comprising the video block in luma samples and (ii) a height of the video block in luma samples is greater than VSize.
15. The method of claim 1, wherein a horizontal BT partitioning is enabled when a sum of a width of the video block in luma samples and a horizontal coordinate of a top-left luma sample of the video block is greater than a width of a picture or a width of a subpicture comprising the video block in luma samples.
16. The method of claim 1, wherein the determination of a horizontal BT partitioning is based on a width of the video block in luma samples being greater than VSize and a height of the video block in luma samples being less than or equal to VSize.
17. The method of claim 16, wherein a maximum transform size is 32×32 and VSize is 64×64 , and wherein the horizontal BT partitioning is disabled when a size of the video block is 128×64 .
18. The method of claim 16, wherein a maximum transform size is 32×32 and VSize is 64×64 , and wherein the horizontal BT partitioning is enabled when a size of the video block is 64×8 , 64×16 , or 64×32 .
19. The method of claim 1, wherein a horizontal BT partitioning is disabled when (i) a sum of a height of the video block in luma samples and a vertical coordinate of a top-left luma sample of the video block is greater than a height of a picture or a height of a subpicture comprising the

video block in luma samples and (ii) a width of the video block in luma samples is greater than VSize.

20. The method of claim 1, wherein a vertical BT partitioning is enabled when a sum of a height of the video block in luma samples and a vertical coordinate of a top-left luma sample of the video block is greater than a height of a picture or a height of a subpicture comprising the video block in luma samples.

21. The method of claim 1, wherein the TT or the BT partitioning is disabled and an indication of the TT or the BT partitioning is excluded from the bitstream representation, and wherein the indication is implicitly derived to be a predetermined value that indicates the TT or the BT partitioning is disabled.

22. The method of claim 21, wherein the predetermined value is zero.

23. The method of claim 1, wherein the TT or the BT partitioning is enabled and an indication of the TT or the BT partitioning is signaled in the bitstream representation.

24. The method of claim 1, wherein the TT or the BT partitioning is disabled, wherein an indication of the TT or the BT partitioning is signaled in the bitstream representation, and wherein the indication is ignored by a decoder.

25. The method of claim 1, wherein the TT or the BT partitioning is disabled, wherein an indication of the TT or the BT partitioning is signaled in the bitstream representation, and wherein the indication is zero based on the TT or the BT partitioning being disabled.

26. A method of video processing, comprising:
using, for a conversion between a video comprising one or more video regions comprising one or more video blocks and a bitstream representation of the video, a dimension of a video block of the one or more video blocks to perform a determination of whether a ternary-tree (TT) or a binary-tree (BT) partitioning of the video block is enabled; and
performing, based on the determination, the conversion.

27. The method of claim 26, wherein the determination of the TT or the BT partitioning is based on a height or a width of the video block in luma samples being greater than N, and wherein N is a positive integer.

28. The method of claim 27, wherein $N = 64$.
29. The method of claim 27 or 28, wherein a maximum transform size is 32×32 , and wherein the TT partitioning is disabled when a size of the video block is 128×128 , 128×64 , or 64×128 .
30. The method of claim 27 or 28, wherein a maximum transform size is 32×32 , and wherein the TT partitioning is enabled when a size of the video block is 64×64 .
31. The method of claim 26, wherein the determination of a vertical BT partitioning is based on a width of the video block in luma samples being less than or equal to N and a height of the video block in luma samples being greater than N , and wherein N is a positive integer.
32. The method of claim 31, wherein $N = 64$.
33. The method of claim 31 or 32, wherein a maximum transform size is 32×32 , and wherein the vertical BT partitioning is disabled when a size of the video block is 64×128 .
34. The method of claim 31 or 32, wherein a maximum transform size is 32×32 , and wherein the vertical BT partitioning is enabled when a size of the video block is 32×64 , 16×64 , or 8×64 .
35. The method of claim 26, wherein a vertical BT partitioning is disabled when (i) a sum of a width of the video block in luma samples and a horizontal coordinate of a top-left luma sample of the video block is greater than a width of a picture or a width of a subpicture comprising the video block in luma samples and (ii) a height of the video block in luma samples is greater than 64.
36. The method of claim 26, wherein a horizontal BT partitioning is enabled when a sum of a width of the video block in luma samples and a horizontal coordinate of a top-left luma sample of the video block is greater than a width of a picture or a width of a subpicture comprising the video block in luma samples.
37. The method of claim 26, wherein the determination of a horizontal BT partitioning is based on a width of the video block in luma samples being greater than N and a height of the video block in luma samples being less than or equal to N , and wherein N is an integer.
38. The method of claim 37, wherein $N = 64$.

39. The method of claim 37 or 38, wherein a maximum transform size is 32×32 , and wherein the horizontal BT partitioning is disabled when a size of the video block is 128×64 .
40. The method of claim 37 or 38, wherein a maximum transform size is 32×32 , and wherein the horizontal BT partitioning is enabled when a size of the video block is 64×8 , 64×16 , or 64×32 .
41. The method of claim 26, wherein a horizontal BT partitioning is disabled when (i) a sum of a height of the video block in luma samples and a vertical coordinate of a luma sample of the video block is greater than a height of a picture or a height of a subpicture comprising the video block in luma samples and (ii) a width of the video block in luma samples is greater than N, and wherein N is a positive integer.
42. The method of claim 26, wherein a vertical BT partitioning is enabled when a sum of a height of the video block in luma samples and a vertical coordinate of a top-left luma sample of the video block is greater than a height of a picture or a height of a subpicture comprising the video block in luma samples.
43. The method of any of claims 1 to 42, wherein the video block corresponds to a coding tree unit (CTU) representing a logical partition used for coding the video into the bitstream representation.
44. The method of any of claims 1 to 43, wherein performing the conversion comprises generating the bitstream representation from the video region.
45. The method of any of claims 1 to 43, wherein performing the conversion comprises generating the video region from the bitstream representation.
46. An apparatus in a video system comprising a processor and a non-transitory memory with instructions thereon, wherein the instructions upon execution by the processor, cause the processor to implement the method recited in one or more of claims 1 to 45.
47. A computer program product stored on a non-transitory computer readable media, the computer program product including program code for carrying out the method recited in one or more of claims 1 to 45.