CLAIMS

1. A method of decoding video data, the method comprising:

decoding (220) one or more syntax elements at a region-tree level of a region-tree of a tree data structure for a coding tree block, CTB, of video data, the region-tree having one or more region-tree nodes including zero or more region-tree non-leaf nodes and one or more region-tree leaf nodes, each of the region-tree non-leaf nodes having four child region-tree nodes;

determining, using the syntax elements at the region-tree level, how the region-tree nodes are split into the child region-tree nodes;

decoding (224) one or more syntax elements at a prediction-tree level for each of the region-tree leaf nodes of one or more prediction trees of the tree data structure for the CTB, the prediction trees each having root nodes corresponding to one or more of the region-tree leaf nodes and one or more prediction-tree nodes including zero or more prediction-tree non-leaf nodes and one or more prediction-tree leaf nodes, each of the prediction-tree non-leaf nodes having either two child prediction tree nodes or three child prediction tree nodes obtained using a center-side triple partitioning, with at least one prediction-tree non-leaf node having three child prediction tree nodes obtained using a center-side triple partitioning, each of the prediction-tree leaf nodes defining respective coding units, CUs;

determining, using the syntax elements at the prediction-tree level, how the prediction-tree nodes are split into the child prediction-tree nodes; and

decoding (228) video data, including prediction data and transform data, for each of the CUs based at least in part on the syntax elements at the region-tree level and the syntax elements at the prediction-tree level, wherein the prediction data indicates a prediction mode for forming a predicted block for a corresponding one of the CUs, and wherein the transform data includes transform coefficients representing transformed residual data for the corresponding one of the CUs.

2. The method of claim 1, wherein decoding the syntax elements at the region-tree level and decoding the syntax elements at the prediction-tree level comprises decoding one or more no-further-split tree types for at least one of the region-tree level or the prediction-tree level, wherein a no-further-split tree type means that no further splitting is permitted.

3. The method of claim 1, further comprising decoding data representing a maximum region-tree depth for the region-tree level.

4. The method of claim 3, further comprising decoding data representing a maximum prediction-tree depth for the prediction-tree level; and preferably

wherein a sum of the maximum region-tree depth and the maximum prediction-tree depth is less than a maximum total depth value.

5. The method of claim 3, wherein decoding the data representing the maximum region-tree depth comprises decoding the data representing the maximum region-tree depth from one or more of a sequence parameter set, SPS, a picture parameter set, PPS, or a slice header.

6. The method of claim 1, further comprising decoding one syntax element that jointly represents both a maximum region-tree depth for the region-tree level and a maximum prediction-tree depth for the prediction-tree level; and preferably

wherein decoding the one syntax element that jointly represents the maximum region-tree depth and the maximum prediction-tree depth comprises decoding the one syntax element from one or more of a sequence parameter set, SPS, a picture parameter set, PPS, or a slice header.

7. The method of claim 1, further comprising inferring that at least one node, comprising at least one of the region-tree nodes or at least one of the prediction-tree nodes, is split without decoding split data for the node; and preferably

wherein inferring comprises inferring that the at least one node is split based on a block to which the node corresponds crossing at least one of picture boundary, a slice boundary, or a tile boundary.

8. The method of claim 1, further comprising decoding data indicating whether the region-tree depth and the prediction-tree depth overlap with each other.

9. The method of claim 1, wherein the video data for each of the CUs comprises one or more of a skip flag, a merge index, a syntax element representing whether the CU is predicted using inter-mode or intra-mode, intra-mode prediction information,

motion information, transform information, residual information, or quantization information.

10. The method of claim 1, wherein decoding the one or more syntax elements at the region-tree level comprises decoding one or more syntax elements representing splitting information for the prediction trees before decoding the video data for each of the CUs; and preferably

wherein the method further comprises determining the number of CUs using the splitting information before decoding the video data for each of the CUs.

11. The method of claim 1, wherein decoding the syntax elements at the region-tree level comprises decoding one or more syntax elements representing one or more enabled coding tools at the region-tree level; and preferably

wherein the method further comprises applying the coding tools across boundaries of the CUs when the boundaries of the CUs are within a common region as indicated by one of the region-tree leaf nodes of the region-tree.

12. The method of claim 11, wherein either (i) decoding the data representing one or more of the enabled coding tools comprises decoding overlapped block motion compensation, OBMC, mode information in each of the region-tree leaf nodes representing whether OBMC is enabled for blocks of the video data corresponding to the one of the region-tree leaf nodes; or

wherein (ii) decoding the data representing one or more of the enabled coding tools comprises decoding overlapped transforms information in each of the region-tree leaf nodes representing whether overlapped transforms are enabled for blocks of the video data corresponding to the one of the region-tree leaf nodes, wherein overlapped transforms comprises a coding tool for which a transform block is permitted to overlap a boundary between two prediction blocks of a region corresponding to the one of the region-tree leaf nodes; or

wherein (iii) decoding the data representing one or more of the enabled coding tools comprises decoding data in one or more of the region-tree leaf nodes indicating whether all of the CUs within a region corresponding to the region-tree leaf node are coded using one of skip mode, merge mode, intra-mode, inter-mode, or frame-rate

upconversion (FRUC) mode, and wherein when all of the CUs within one of the regions are coded using a common mode, the method further comprises preventing decoding of mode information for the CUs at the CU level.

13. The method of claim 1, wherein either (i) decoding the syntax elements at the region-tree level comprises decoding at least one of sample adaptive offset, SAO, parameters or adaptive loop filter, ALF, parameters; or

Wherein (ii) the method further comprises decoding one or more center-side triple tree syntax elements in at least one of the region-tree level or the prediction-tree level; or

Wherein (iii) the method further comprises calculating respective quantization parameters, QPs for each of the CUs, wherein calculating the respective QPs comprises determining base QPs for each of the region-tree leaf nodes and calculating the respective QPs based on the base QPs of the corresponding region-tree leaf nodes of the CUs.

14. The method of claim 1, further comprising encoding the video data prior to decoding the video data.

15. A device (14) for decoding video data, the device comprising:
a memory (64) configured to store video data; and
a processor implemented in circuitry and configured to:
decode one or more syntax elements at a region-tree level of a region-tree of a tree data structure for a coding tree block, CTB, of video data, the region-tree having one or more region-tree nodes including zero or more region-tree non-leaf nodes and one or more region-tree leaf nodes, each of the region-tree non-leaf nodes having four region-tree nodes;
determine, using the syntax elements at the region-tree level, how the region-tree nodes are split into the child region-tree nodes;
decode one or more syntax elements at a prediction-tree level for each of the region-tree leaf nodes of one or more prediction trees of the tree data structure for the CTB, the prediction trees each having root nodes corresponding to one or more of the region-tree leaf nodes and one or more prediction-tree nodes including zero or more prediction-tree non-leaf nodes and one or more

prediction-tree leaf nodes, each of the prediction-tree non-leaf nodes having either two child prediction tree nodes or three child prediction tree nodes obtained using a center-side triple partitioning, with at least one prediction-tree non-leaf node having three child prediction tree nodes obtained using a center-side triple partitioning, each of the prediction-tree leaf nodes defining respective coding units, CUs;

determine, using the syntax elements at the prediction-tree level, how the prediction-tree nodes are split into the child prediction-tree nodes; and

decode video data, including prediction data and transform data, for each of the CUs based at least in part on the syntax elements at the region-tree level and the syntax elements at the prediction-tree level, wherein the prediction data indicates a prediction mode for forming a predicted block for a corresponding one of the CUs, and wherein the transform data includes transform coefficients representing transformed residual data for the corresponding one of the CUs.

16. The device of claim 15, further comprising at least one of:
a display configured to display the decoded video data; or
a camera configured to capture the video data.

17. The device of claim 15, wherein the device comprises one or more of a camera, a computer, a mobile device, a broadcast receiver device, or a set-top box.

18. A non-transitory computer-readable storage medium having stored thereon instructions that, when executed, cause a processor to carry out the method steps of any one of claims 1 to 14.