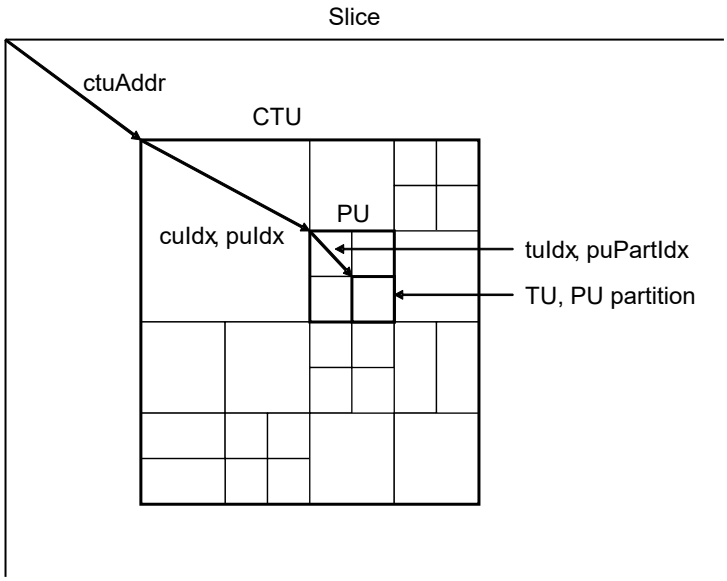

Figure 1

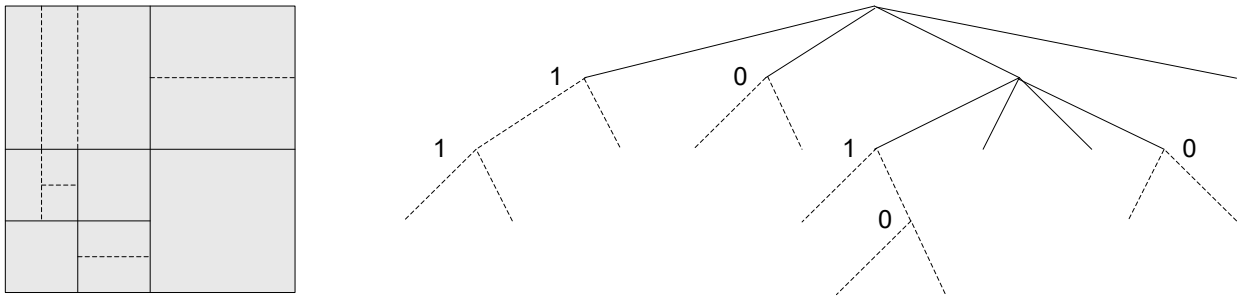

Figure 2

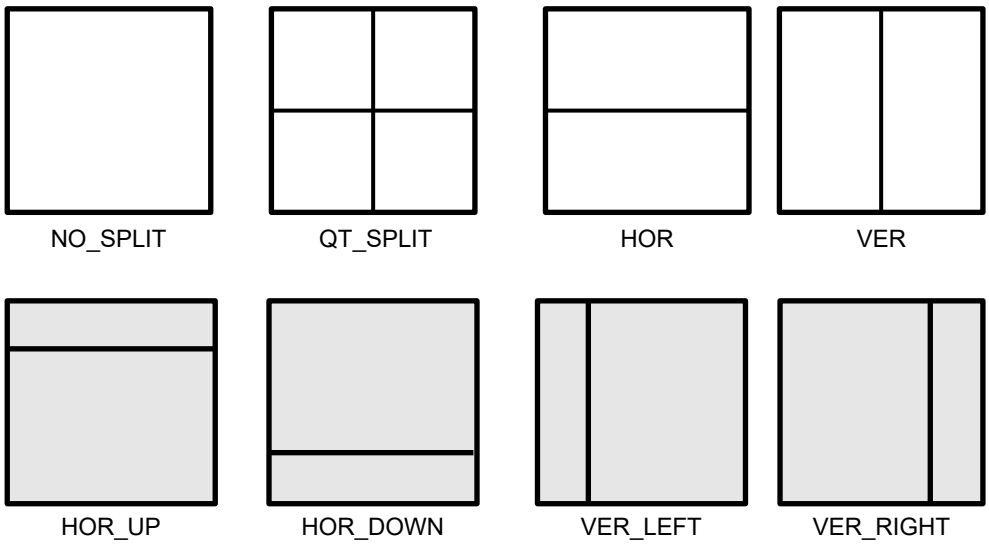

Figure 3

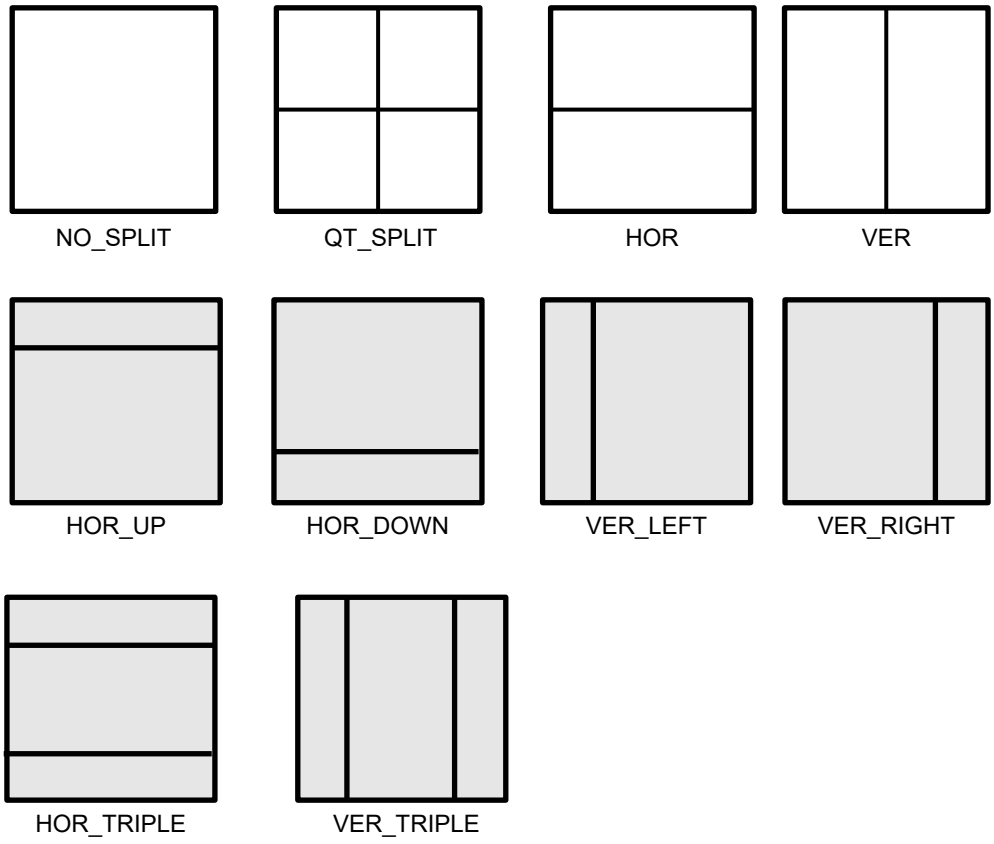

Figure 4

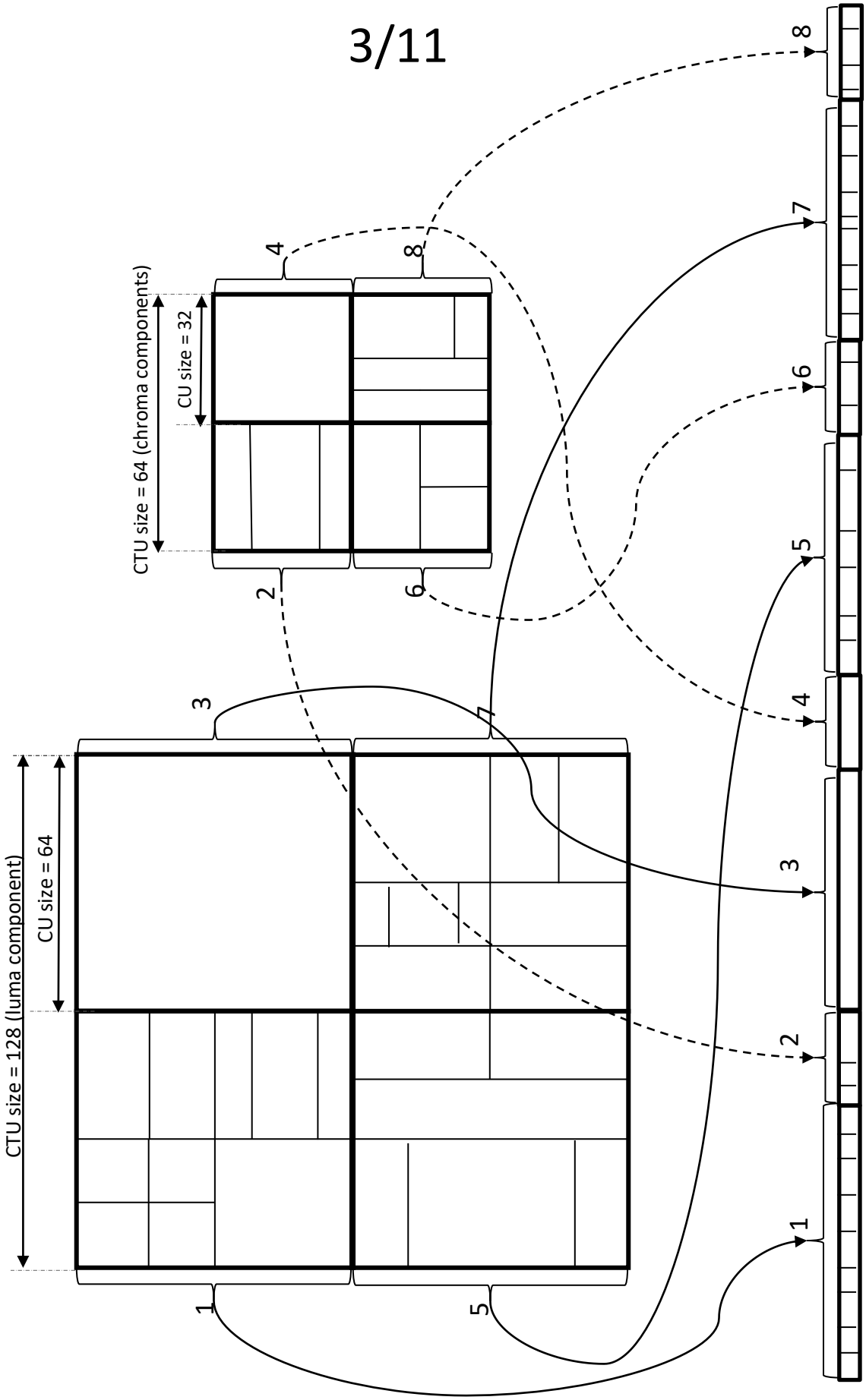
CTU size = 64 (chroma components)
CU size = 32
CTU size = 128 (luma component)
CU size = 64
1
2
3
4
5
6
7
8

Figure 5

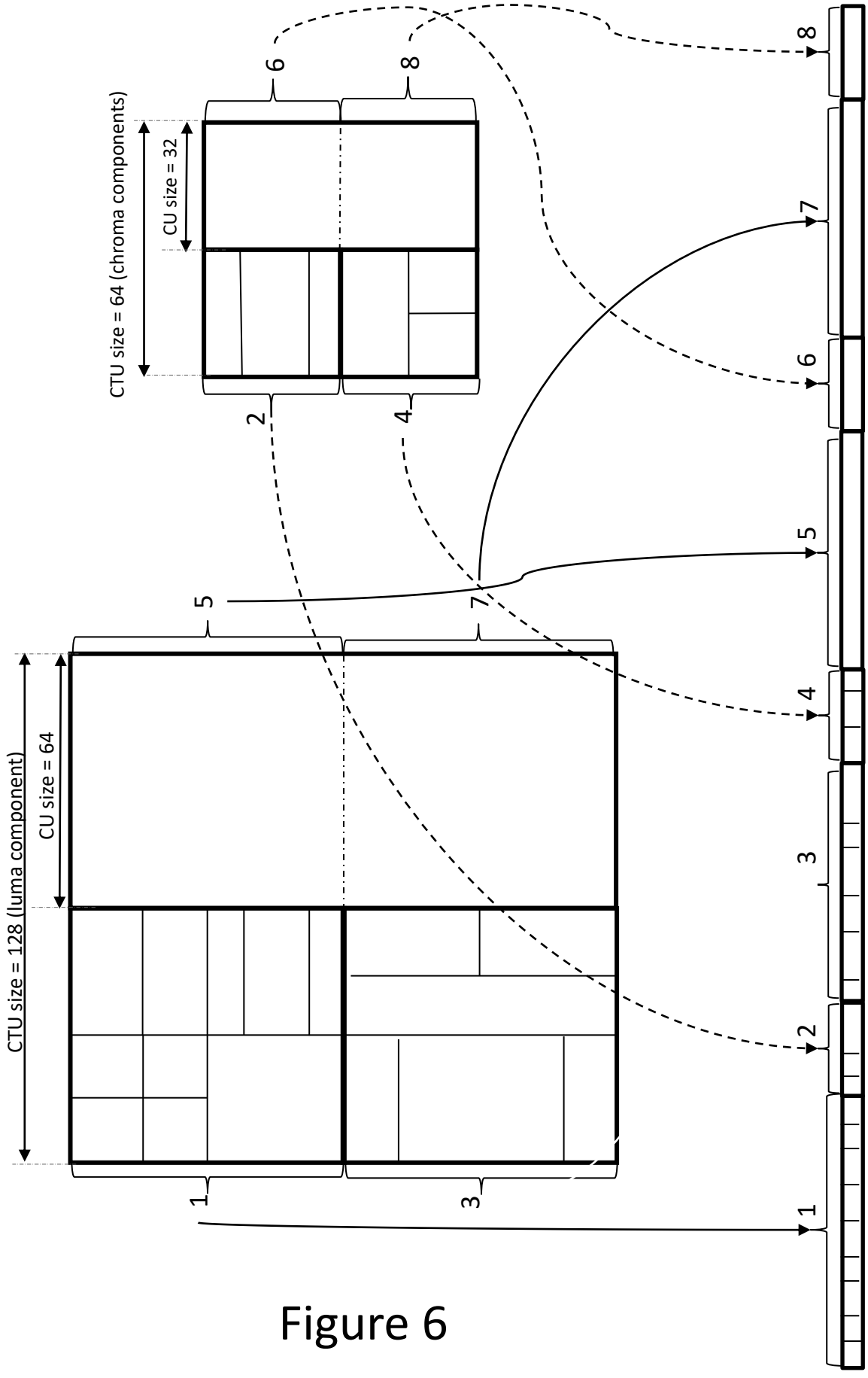
CTU size = 64 (chroma components)
CU size = 32
CTU size = 128 (luma component)
CU size = 64
1
2
3
4
5
6
7
8

Figure 6

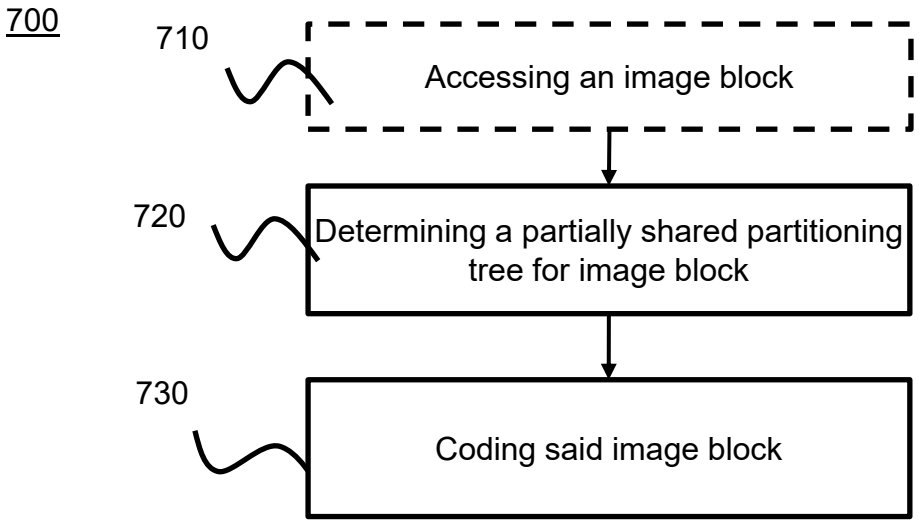

Figure 7

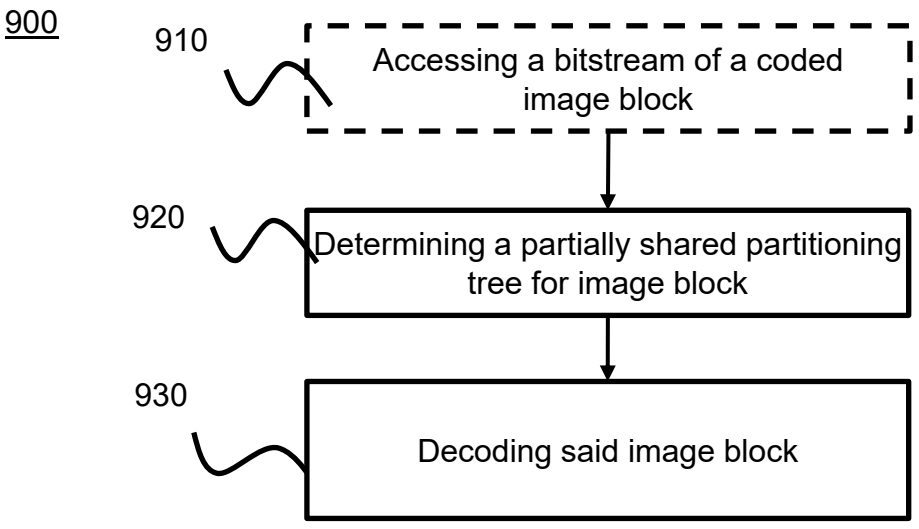

Figure 9

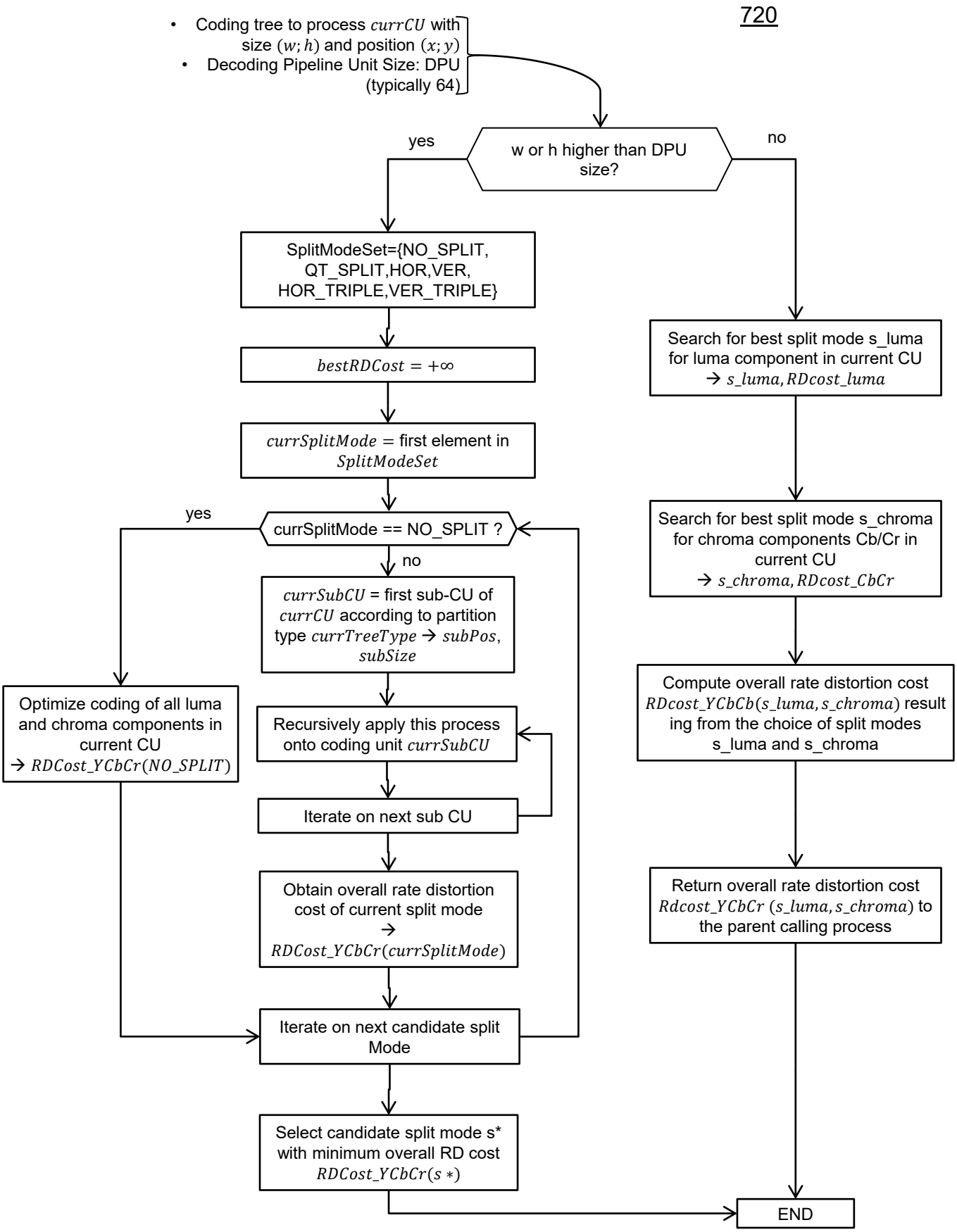
720
• Coding tree to process $currCU$ with size $(w; h)$ and position $(x; y)$
• Decoding Pipeline Unit Size: DPU (typically 64)
w or h higher than DPU size?
yes
no
SplitModeSet={NO_SPLIT, QT_SPLIT,HOR,VER, HOR_TRIPLE,VER_TRIPLE}
$bestRDCost = +\infty$
$currSplitMode$ = first element in $SplitModeSet$
currSplitMode == NO_SPLIT ?
yes
no
$currSubCU$ = first sub-CU of $currCU$ according to partition type $currTreeType$ → $subPos$, $subSize$
Recursively apply this process onto coding unit $currSubCU$
Iterate on next sub CU
Obtain overall rate distortion cost of current split mode → $RDCost_YCbCr(currSplitMode)$
Iterate on next candidate split Mode
Select candidate split mode s* with minimum overall RD cost $RDCost_YCbCr(s*)$
Optimize coding of all luma and chroma components in current CU → $RDCost_YCbCr(NO_SPLIT)$
Search for best split mode s_luma for luma component in current CU → $s_luma, RDcost_luma$
Search for best split mode s_chroma for chroma components Cb/Cr in current CU → $s_chroma, RDcost_CbCr$
Compute overall rate distortion cost $RDcost_YCbCb(s_luma, s_chroma)$ resulting from the choice of split modes s_luma and s_chroma
Return overall rate distortion cost $Rdcost_YCbCr\ (s_luma, s_chroma)$ to the parent calling process
END

Figure 8

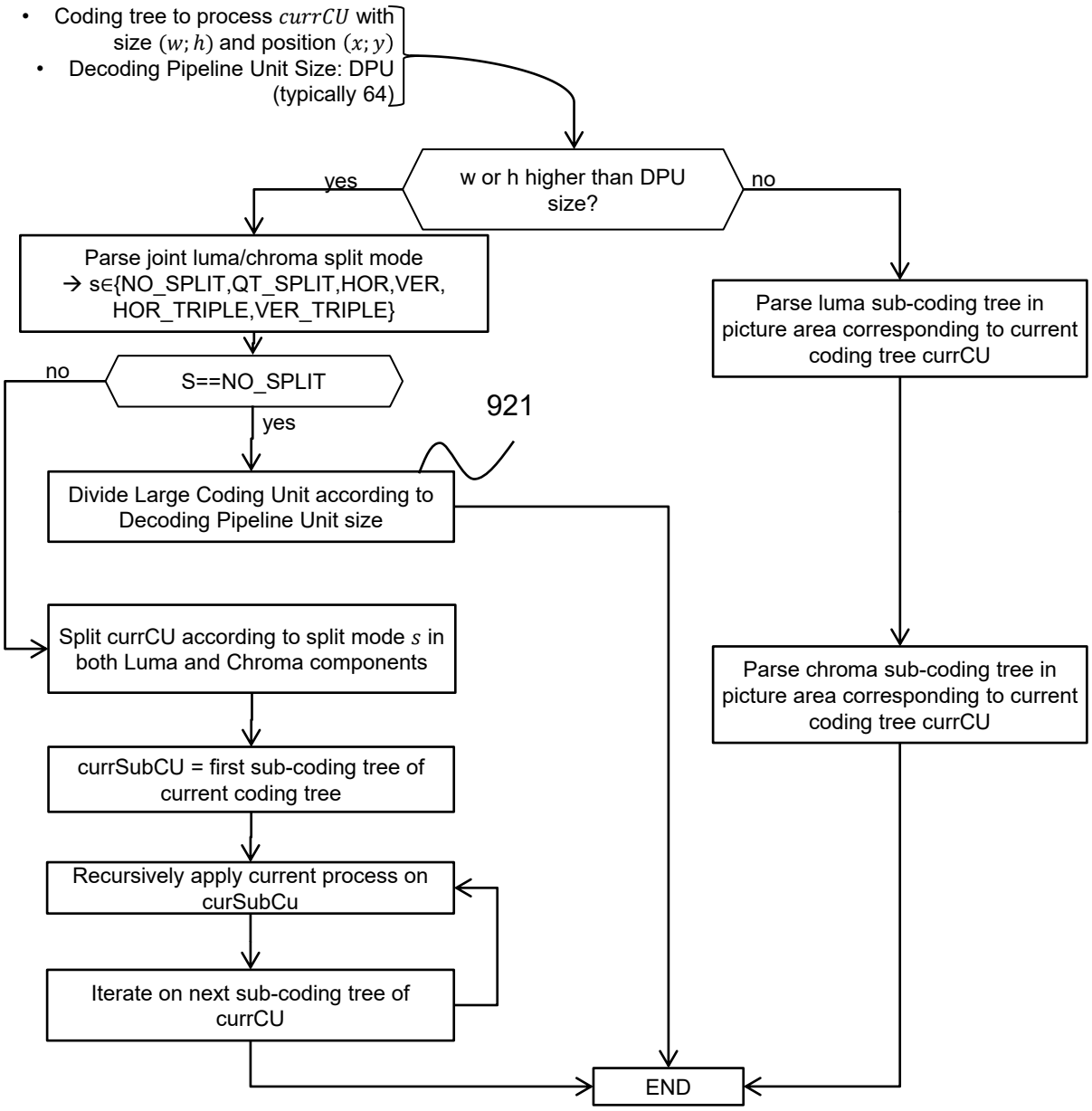
• Coding tree to process $currCU$ with size $(w; h)$ and position $(x; y)$
• Decoding Pipeline Unit Size: DPU (typically 64)
w or h higher than DPU size?
yes
no
Parse joint luma/chroma split mode → s∈{NO_SPLIT,QT_SPLIT,HOR,VER, HOR_TRIPLE,VER_TRIPLE}
Parse luma sub-coding tree in picture area corresponding to current coding tree currCU
S==NO_SPLIT
no
yes
921
Divide Large Coding Unit according to Decoding Pipeline Unit size
Split currCU according to split mode s in both Luma and Chroma components
Parse chroma sub-coding tree in picture area corresponding to current coding tree currCU
currSubCU = first sub-coding tree of current coding tree
Recursively apply current process on curSubCu
Iterate on next sub-coding tree of currCU
END

Figure 10

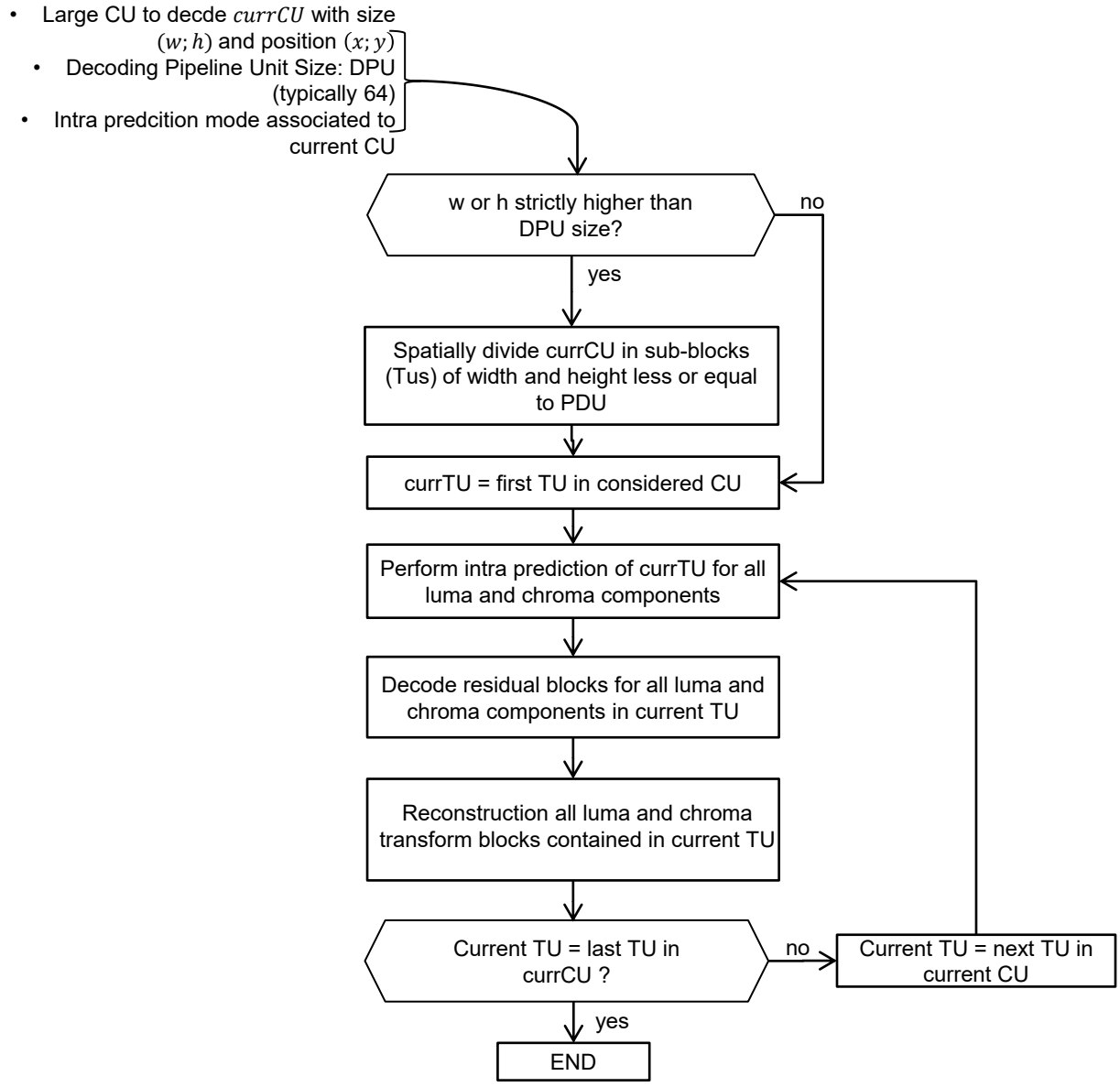
• Large CU to decde $currCU$ with size $(w; h)$ and position $(x; y)$
• Decoding Pipeline Unit Size: DPU (typically 64)
• Intra predcition mode associated to current CU
w or h strictly higher than DPU size?
no
yes
Spatially divide currCU in sub-blocks (Tus) of width and height less or equal to PDU
currTU = first TU in considered CU
Perform intra prediction of currTU for all luma and chroma components
Decode residual blocks for all luma and chroma components in current TU
Reconstruction all luma and chroma transform blocks contained in current TU
Current TU = last TU in currCU ?
no
Current TU = next TU in current CU
yes
END

Figure 11

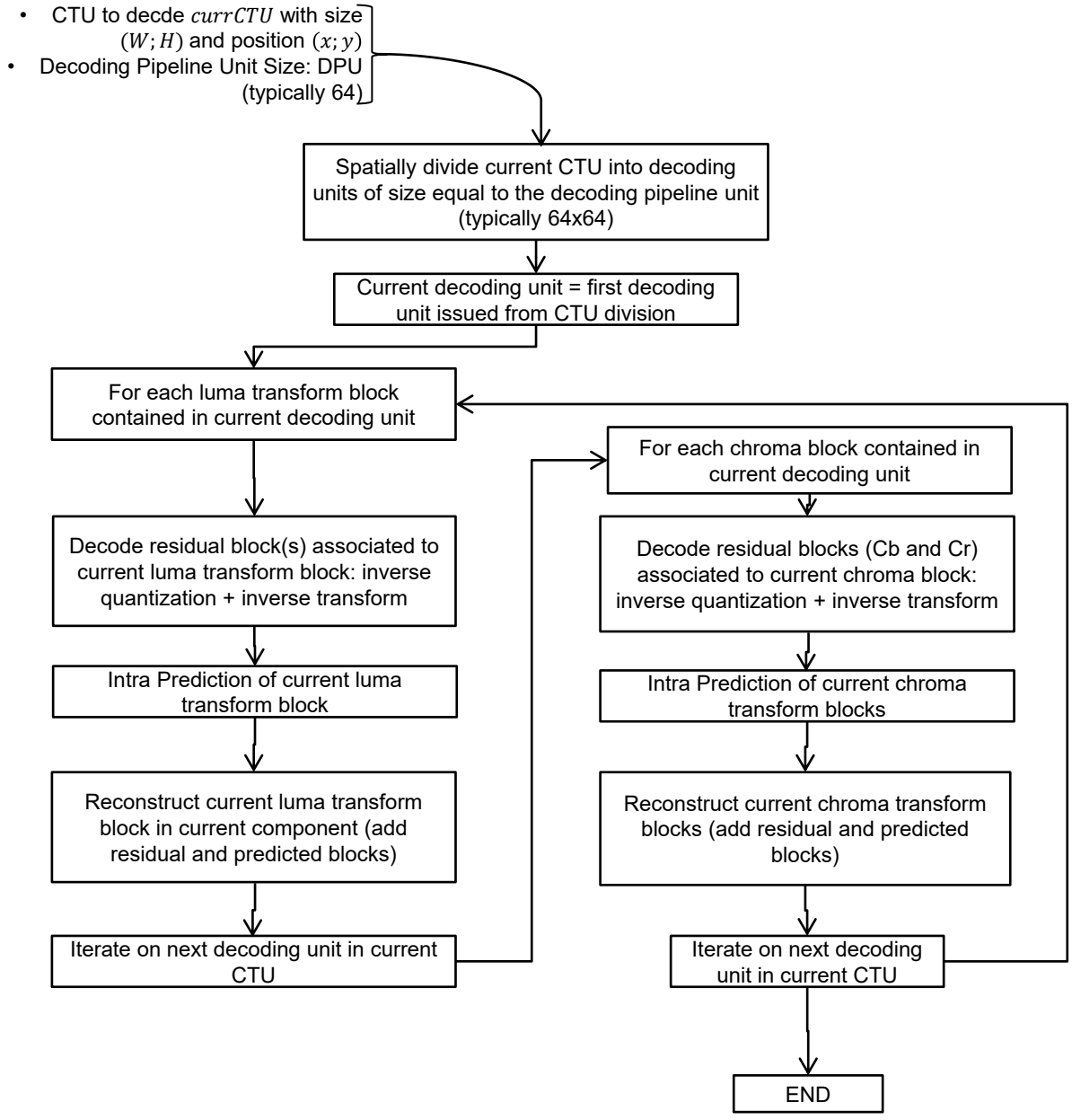
• CTU to decde $currCTU$ with size $(W; H)$ and position $(x; y)$
• Decoding Pipeline Unit Size: DPU (typically 64)
Spatially divide current CTU into decoding units of size equal to the decoding pipeline unit (typically 64x64)
Current decoding unit = first decoding unit issued from CTU division
For each luma transform block contained in current decoding unit
Decode residual block(s) associated to current luma transform block: inverse quantization + inverse transform
Intra Prediction of current luma transform block
Reconstruct current luma transform block in current component (add residual and predicted blocks)
Iterate on next decoding unit in current CTU
For each chroma block contained in current decoding unit
Decode residual blocks (Cb and Cr) associated to current chroma block: inverse quantization + inverse transform
Intra Prediction of current chroma transform blocks
Reconstruct current chroma transform blocks (add residual and predicted blocks)
Iterate on next decoding unit in current CTU
END

Figure 12

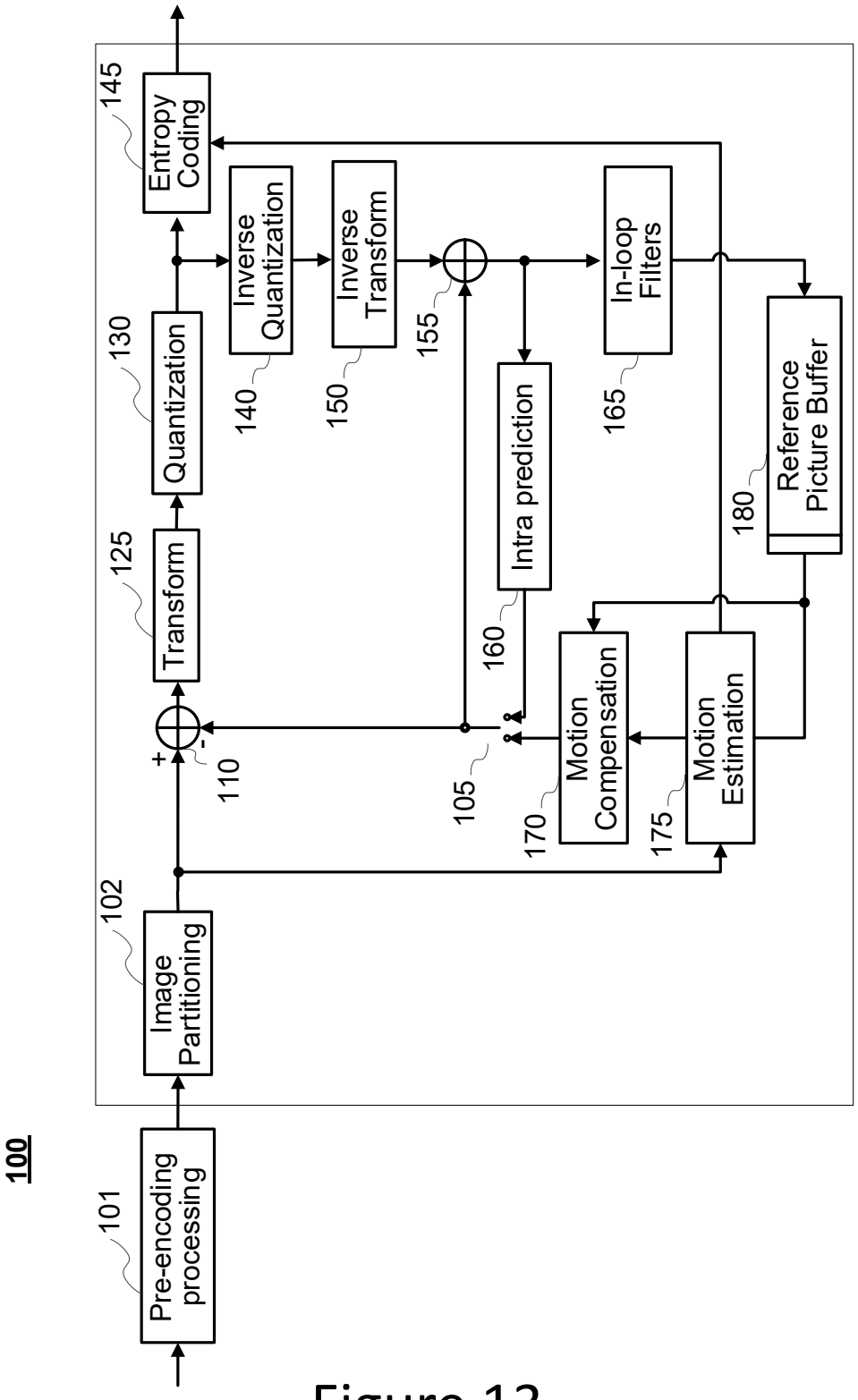
100
101
Pre-encoding processing
102
Image Partitioning
110
+
-
125
Transform
130
Quantization
140
Inverse Quantization
145
Entropy Coding
150
Inverse Transform
155
160
Intra prediction
165
In-loop Filters
170
Motion Compensation
175
Motion Estimation
180
Reference Picture Buffer
105

Figure 13

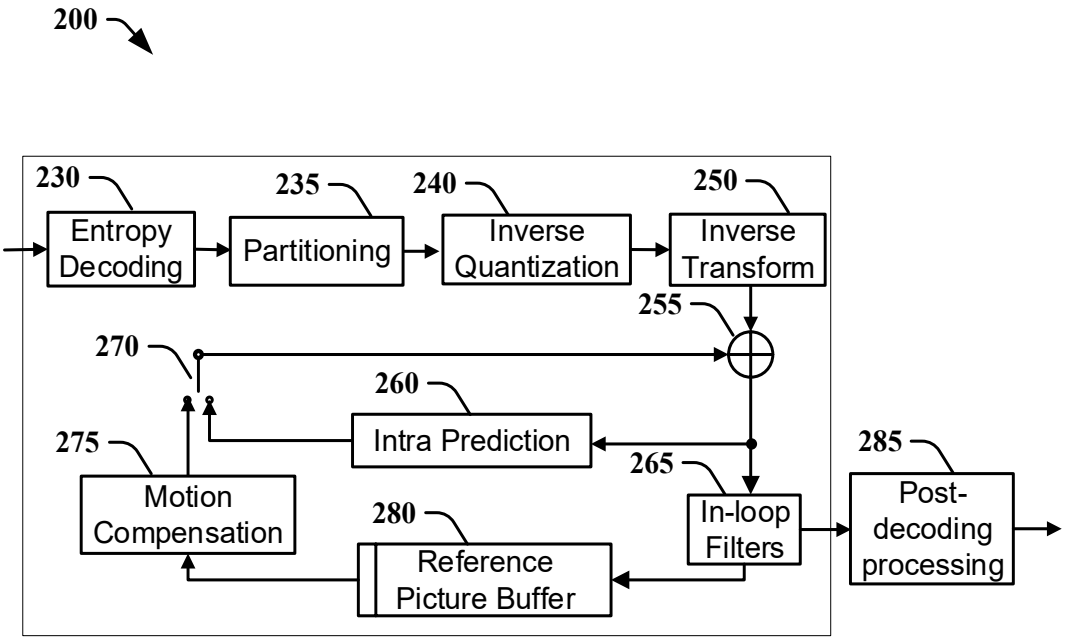

Figure 14

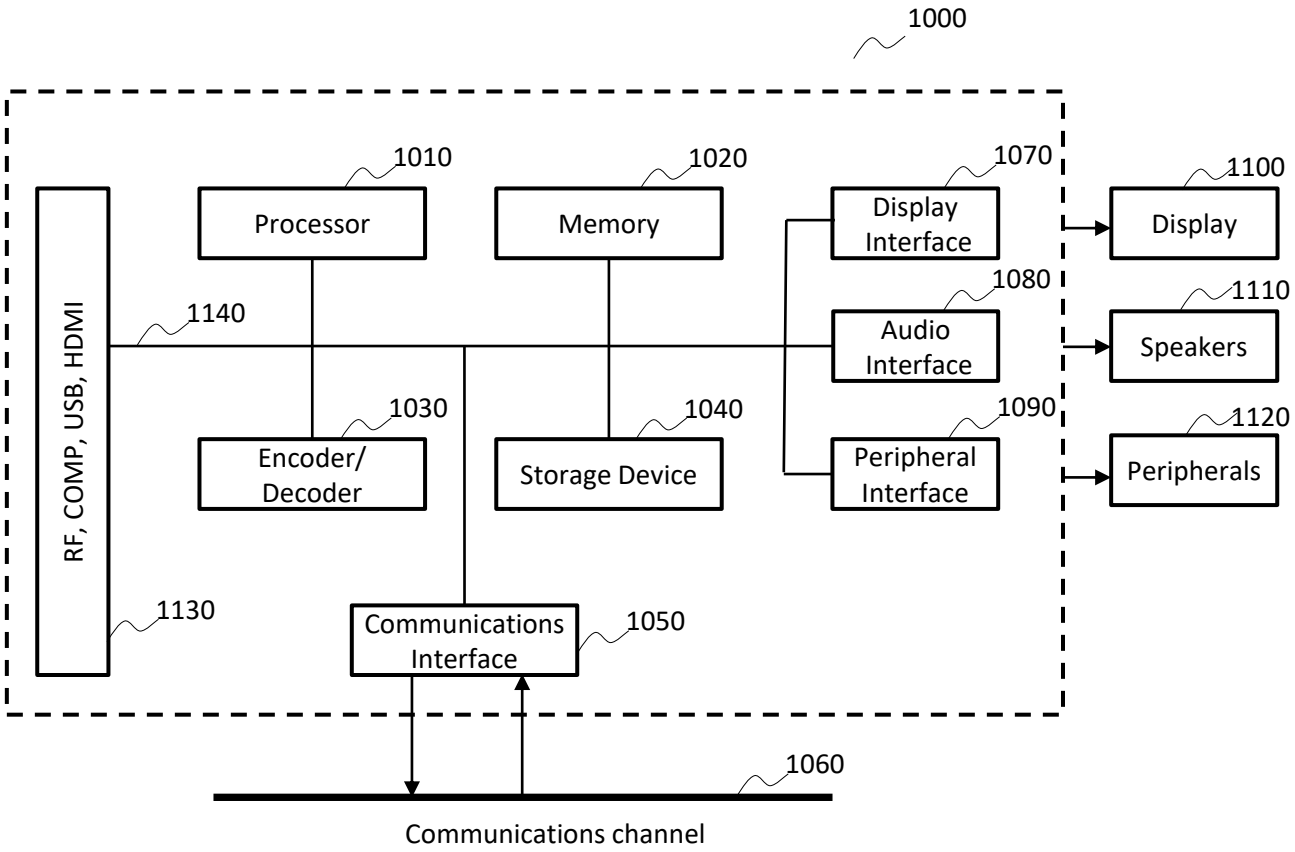

Figure 15