Figure 1
Prior Art

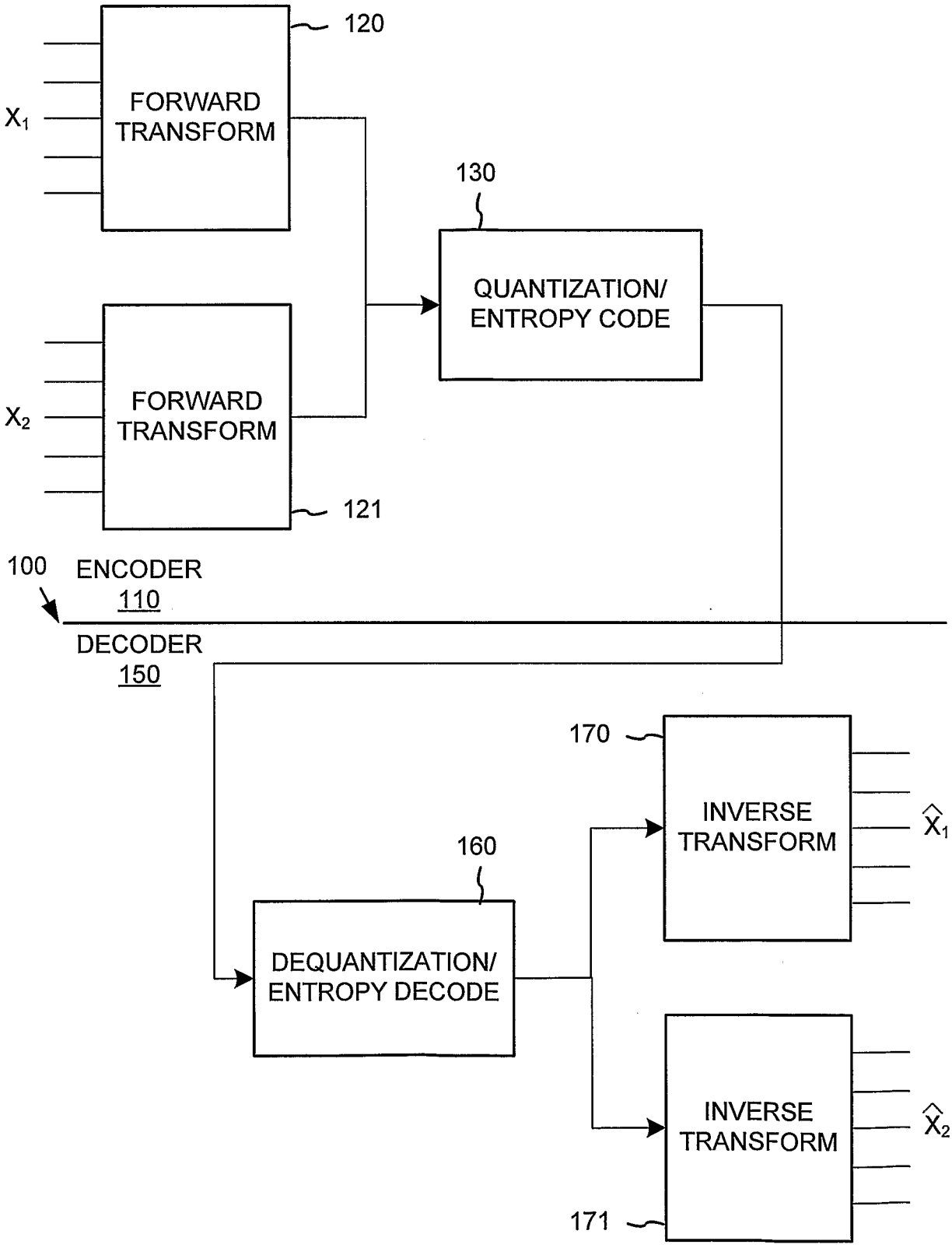


Figure 2

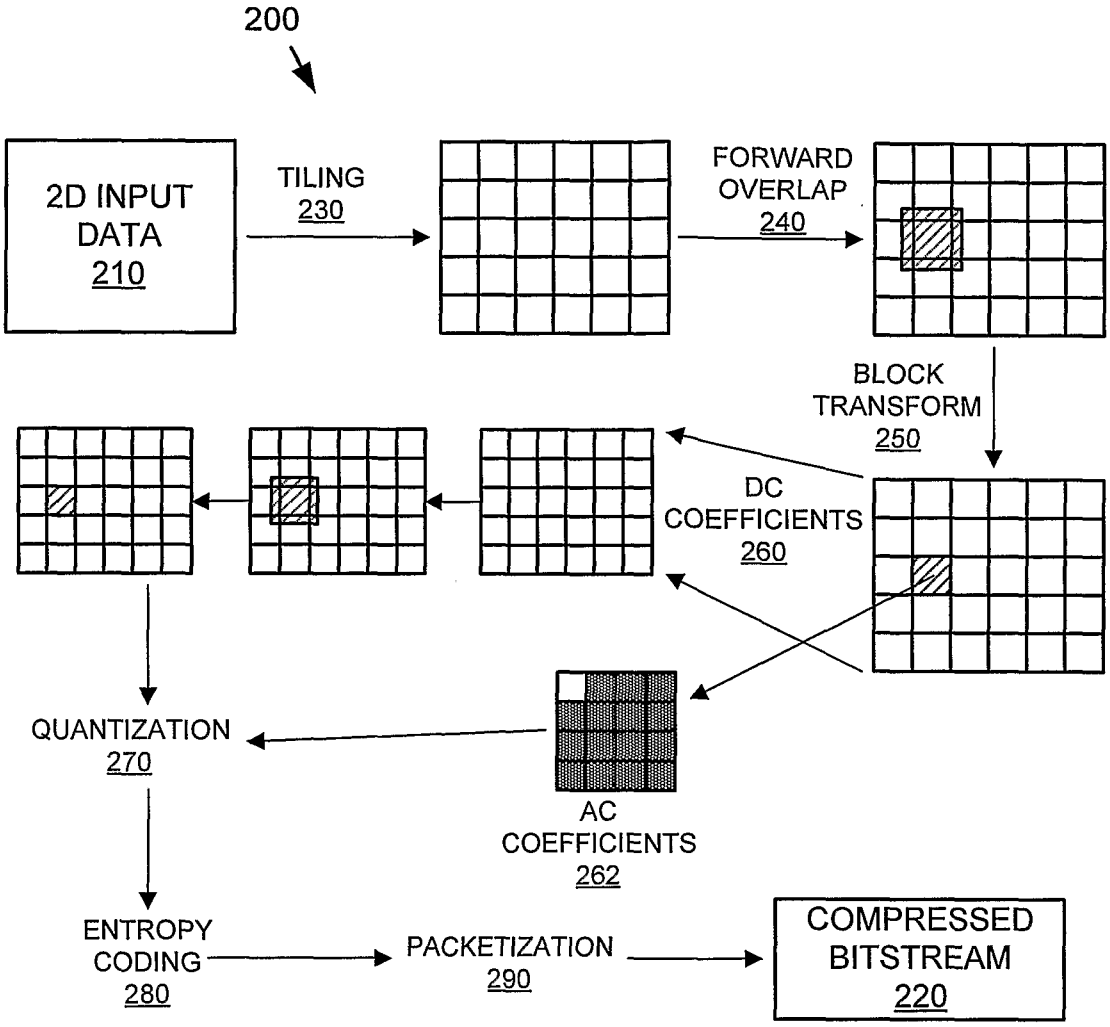

200
2D INPUT DATA 210
TILING 230
FORWARD OVERLAP 240
BLOCK TRANSFORM 250
DC COEFFICIENTS 260
QUANTIZATION 270
AC COEFFICIENTS 262
ENTROPY CODING 280
PACKETIZATION 290
COMPRESSED BITSTREAM 220

Figure 3

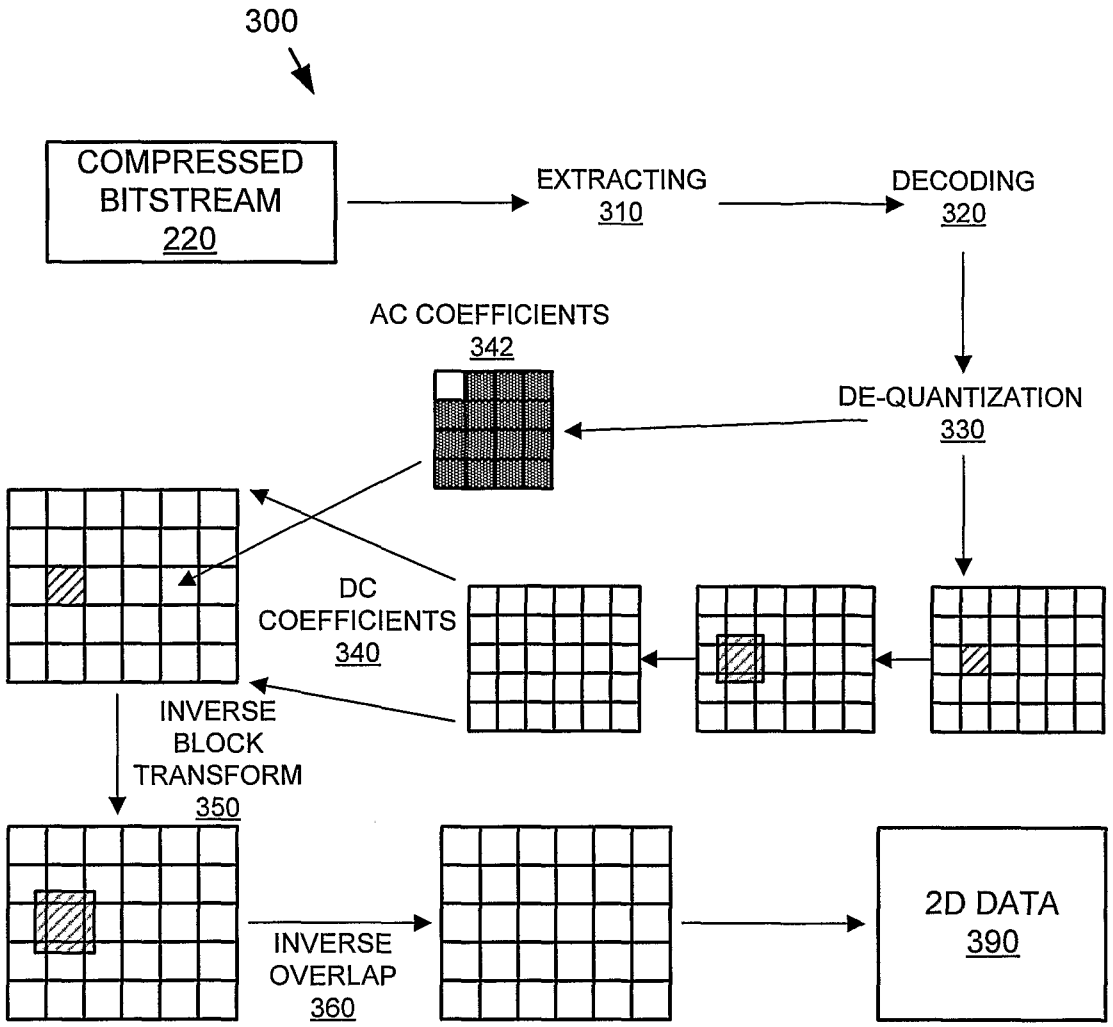

300
COMPRESSED BITSTREAM 220
EXTRACTING 310
DECODING 320
AC COEFFICIENTS 342
DE-QUANTIZATION 330
DC COEFFICIENTS 340
INVERSE BLOCK TRANSFORM 350
INVERSE OVERLAP 360
2D DATA 390

Figure 4

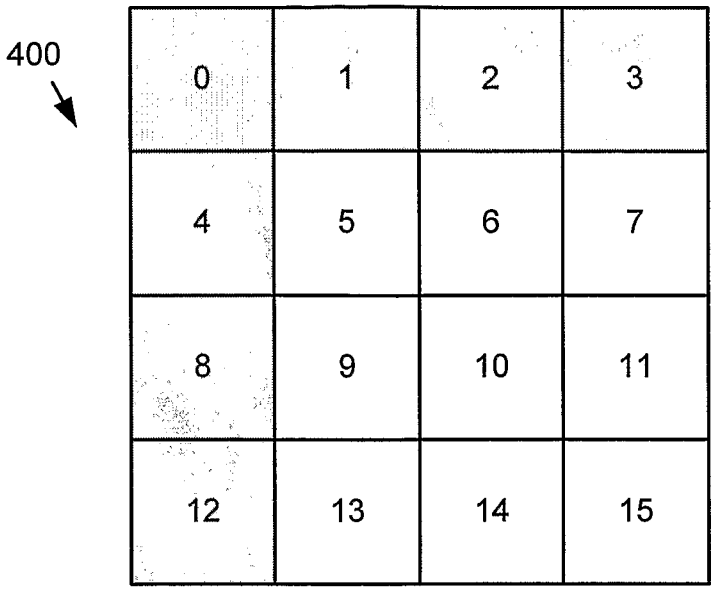


Figure 5

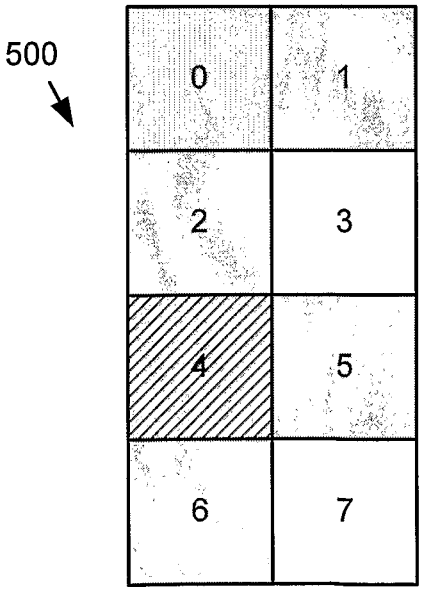


Figure 6

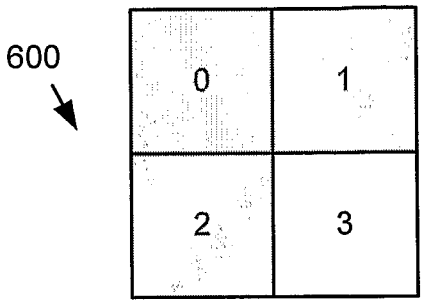


Figure 7

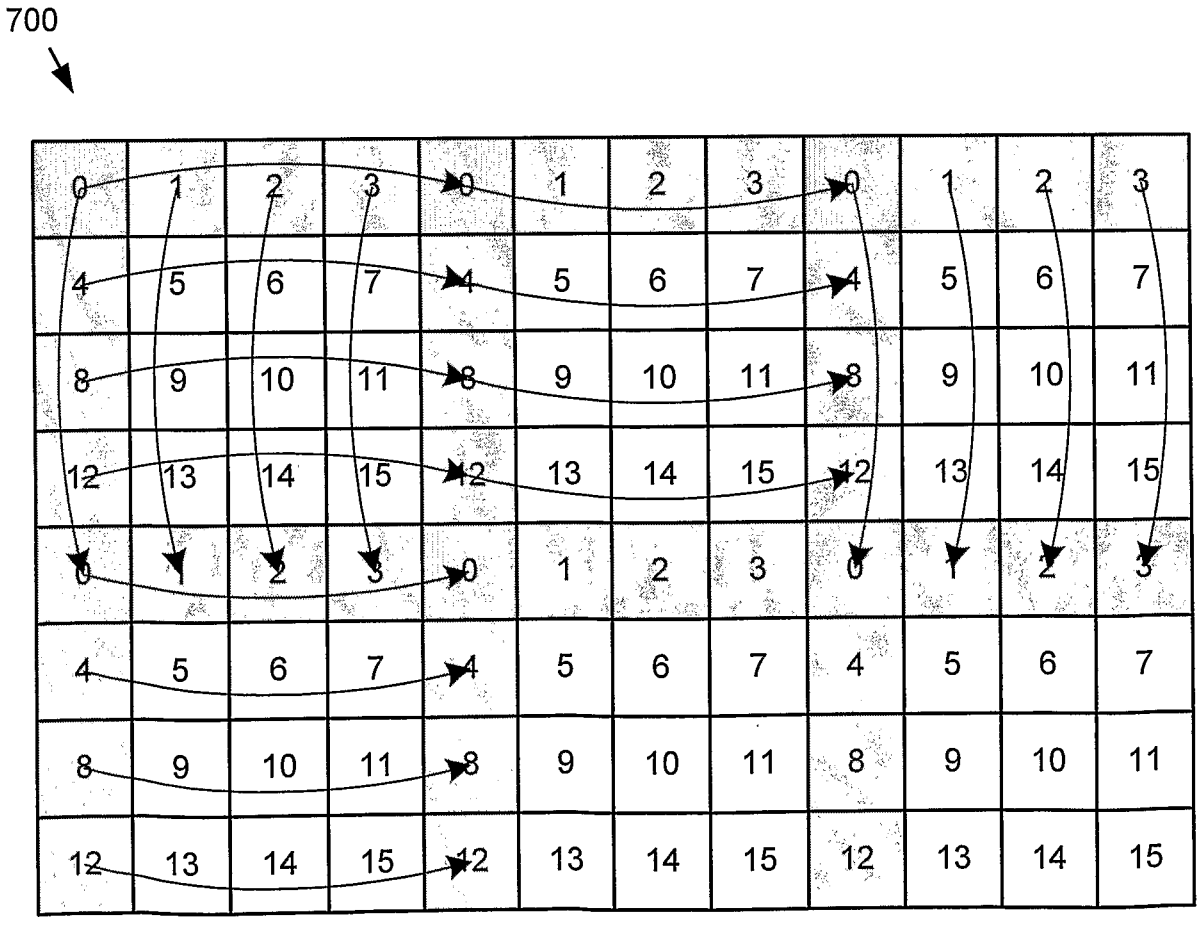


Figure 8

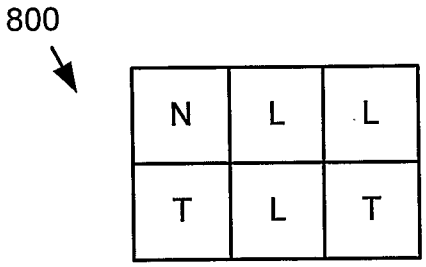


Figure 9

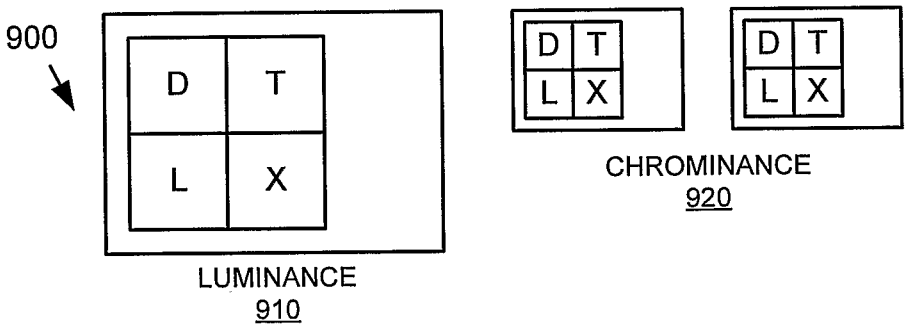
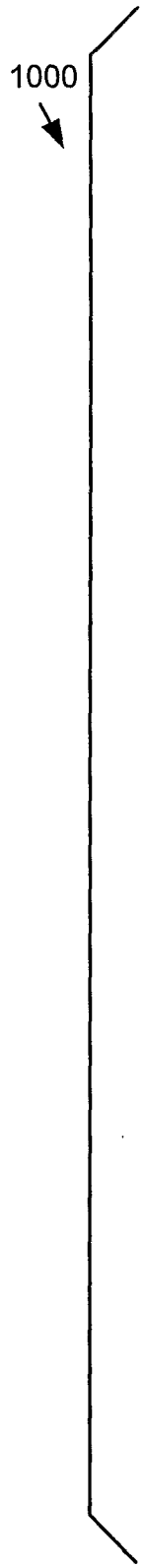


Figure 10




```
If (mx == 0 and my == 0)
     dc_mode = Null predict
Else if (mx == 0)
     dc_mode = Predict from top
Else if (my == 0)
     dc_mode = Predict from left
Else {
     diff_h = abs (D - L)    // luminance
     diff_v = abs (D - T)    // luminance
     If (chrominance channels are available) {
          diff_h = diff_h * scale +
sum_over_chrominance_channels { abs (D - L) }
          diff_v = diff_v * scale +
sum_over_chrominance_channels { abs (D - T) }
     }

     If (diff_h * orient_weight < diff_v) {
          dc_mode = Predict from top
     }
     Else if (diff_v * orient_weight < diff_h) {
          dc_mode = Predict from left
     }
     Else {
          dc_mode = Predict from left and top
     }
}

Where
scale = 8 for YUV 420, 4 for YUV 422, 2 otherwise
orient_weight = 4
```

Figure 11

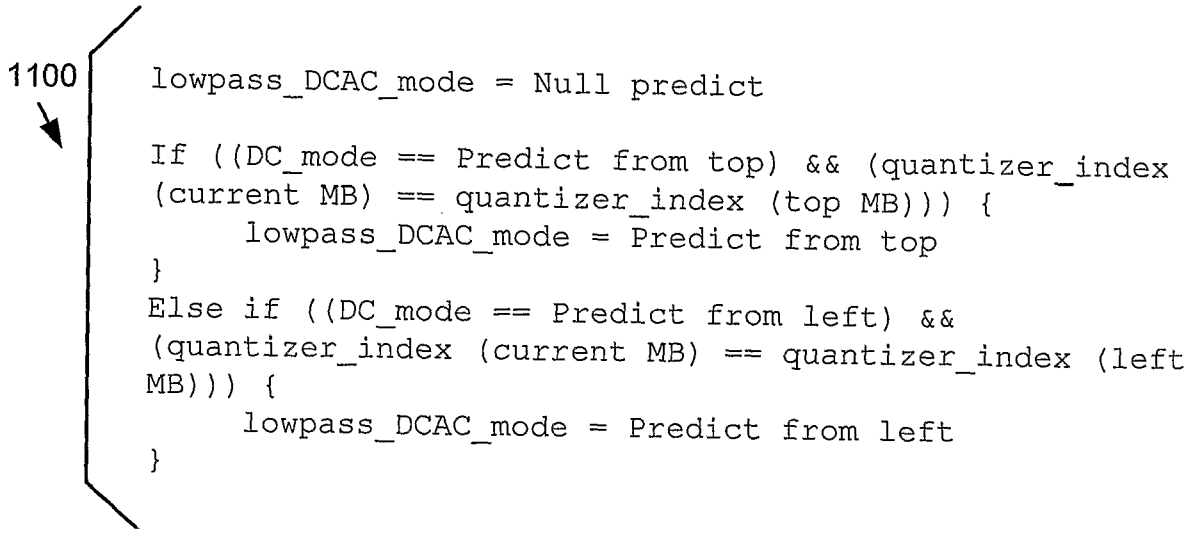

1100
lowpass_DCAC_mode = Null predict
If ((DC_mode == Predict from top) && (quantizer_index (current MB) == quantizer_index (top MB))) {
lowpass_DCAC_mode = Predict from top
}
Else if ((DC_mode == Predict from left) && (quantizer_index (current MB) == quantizer_index (left MB))) {
lowpass_DCAC_mode = Predict from left
}

Figure 12

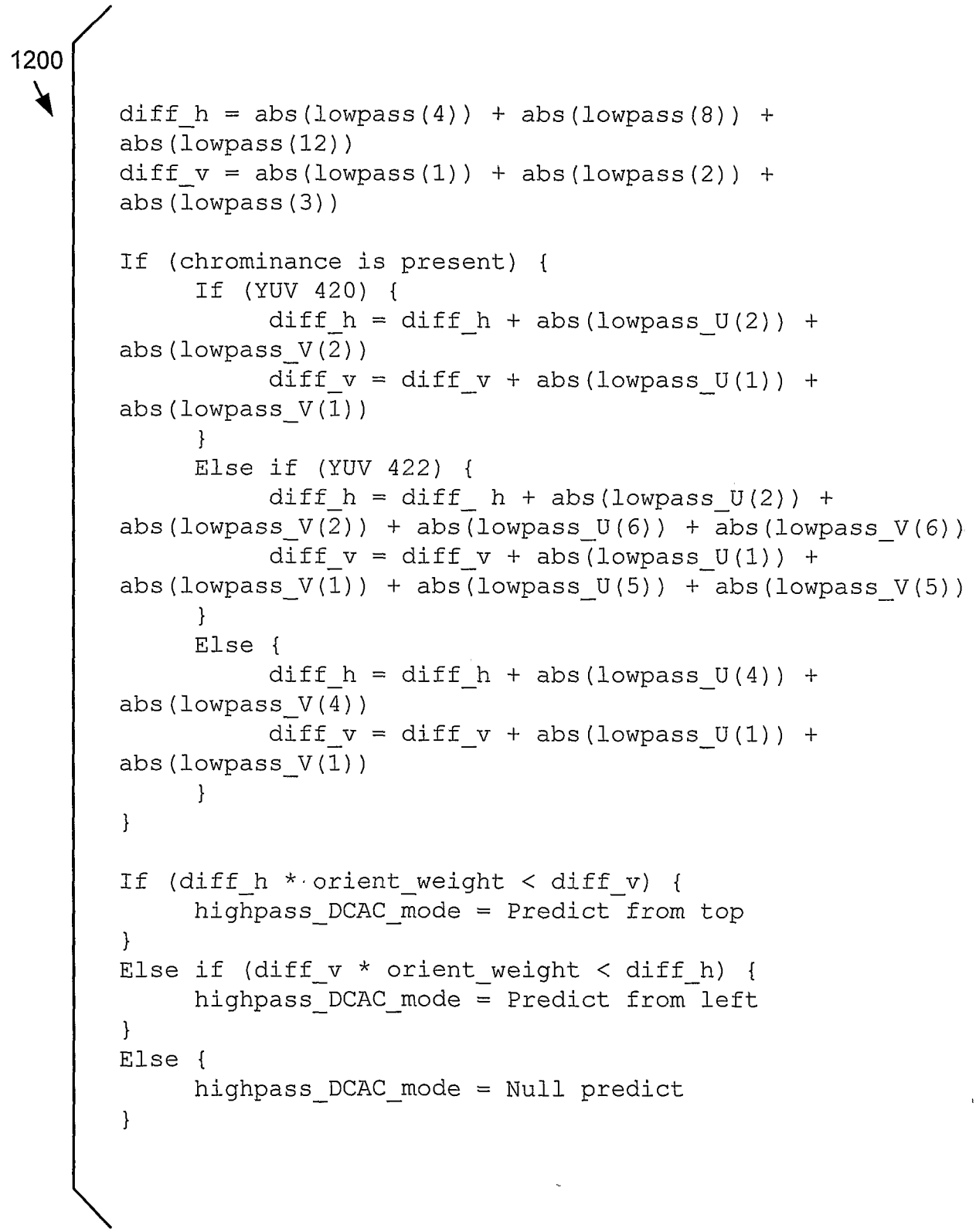

1200
diff_h = abs(lowpass(4)) + abs(lowpass(8)) +
abs(lowpass(12))
diff_v = abs(lowpass(1)) + abs(lowpass(2)) +
abs(lowpass(3))
If (chrominance is present) {
If (YUV 420) {
diff_h = diff_h + abs(lowpass_U(2)) +
abs(lowpass_V(2))
diff_v = diff_v + abs(lowpass_U(1)) +
abs(lowpass_V(1))
}
Else if (YUV 422) {
diff_h = diff_ h + abs(lowpass_U(2)) +
abs(lowpass_V(2)) + abs(lowpass_U(6)) + abs(lowpass_V(6))
diff_v = diff_v + abs(lowpass_U(1)) +
abs(lowpass_V(1)) + abs(lowpass_U(5)) + abs(lowpass_V(5))
}
Else {
diff_h = diff_h + abs(lowpass_U(4)) +
abs(lowpass_V(4))
diff_v = diff_v + abs(lowpass_U(1)) +
abs(lowpass_V(1))
}
}
If (diff_h * orient_weight < diff_v) {
highpass_DCAC_mode = Predict from top
}
Else if (diff_v * orient_weight < diff_h) {
highpass_DCAC_mode = Predict from left
}
Else {
highpass_DCAC_mode = Null predict
}

Figure 13

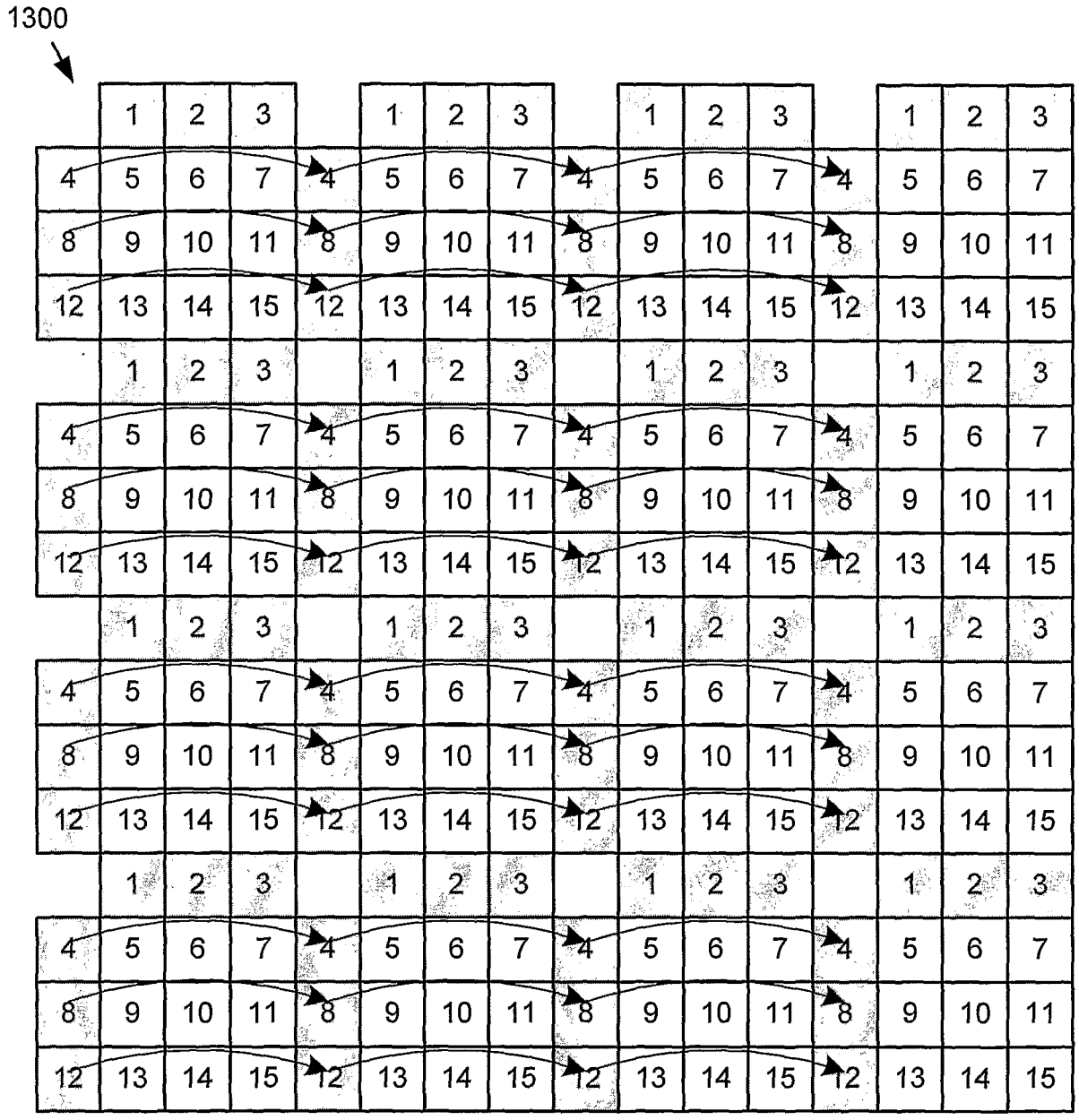

1300
1 2 3 1 2 3 1 2 3 1 2 3
4 5 6 7 4 5 6 7 4 5 6 7 4 5 6 7
8 9 10 11 8 9 10 11 8 9 10 11 8 9 10 11
12 13 14 15 12 13 14 15 12 13 14 15 12 13 14 15
1 2 3 1 2 3 1 2 3 1 2 3
4 5 6 7 4 5 6 7 4 5 6 7 4 5 6 7
8 9 10 11 8 9 10 11 8 9 10 11 8 9 10 11
12 13 14 15 12 13 14 15 12 13 14 15 12 13 14 15
1 2 3 1 2 3 1 2 3 1 2 3
4 5 6 7 4 5 6 7 4 5 6 7 4 5 6 7
8 9 10 11 8 9 10 11 8 9 10 11 8 9 10 11
12 13 14 15 12 13 14 15 12 13 14 15 12 13 14 15
1 2 3 1 2 3 1 2 3 1 2 3
4 5 6 7 4 5 6 7 4 5 6 7 4 5 6 7
8 9 10 11 8 9 10 11 8 9 10 11 8 9 10 11
12 13 14 15 12 13 14 15 12 13 14 15 12 13 14 15

Figure 14

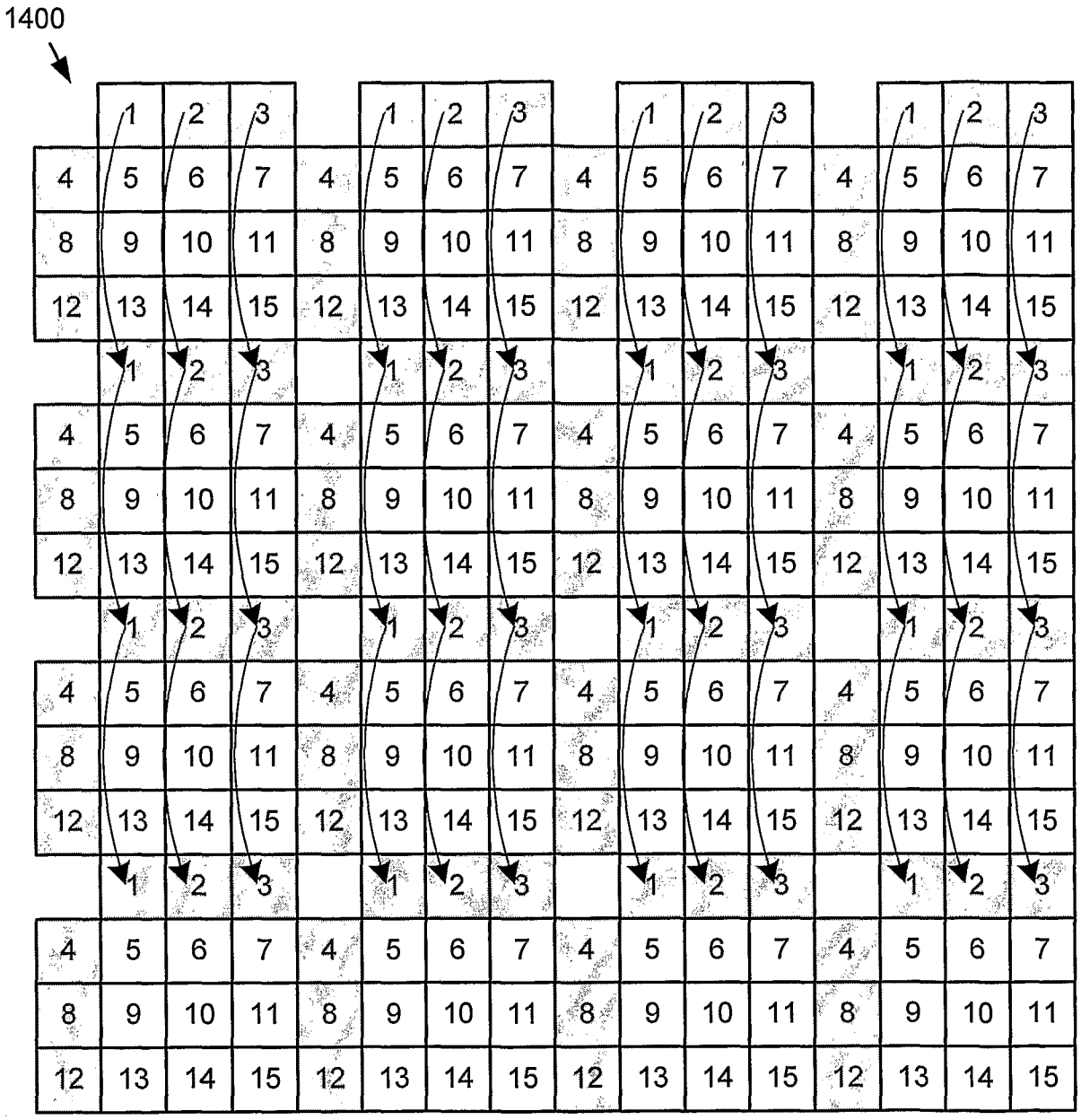
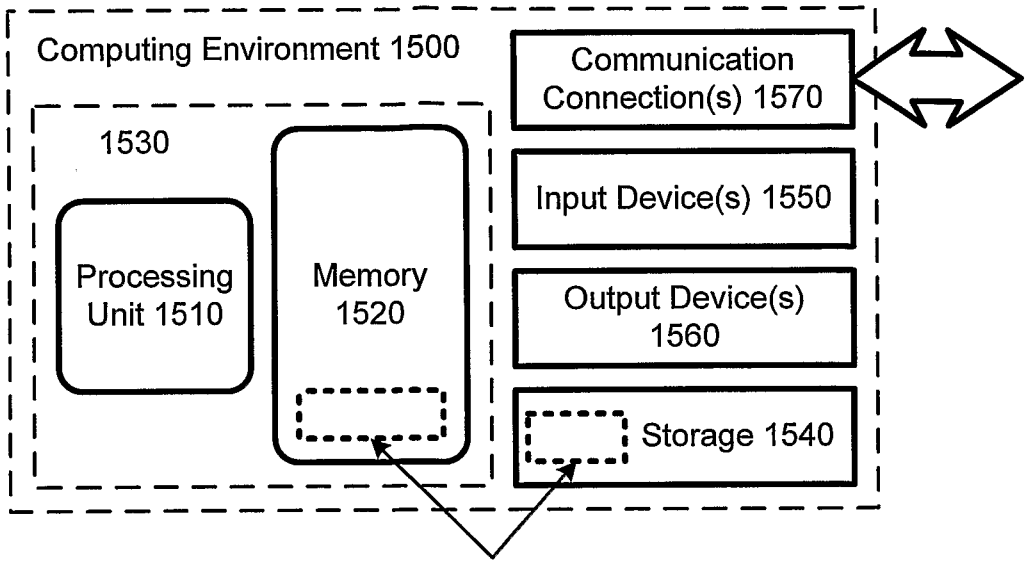


Figure 15



Software 1580 Implementing Predictive Transform Coefficient Coding