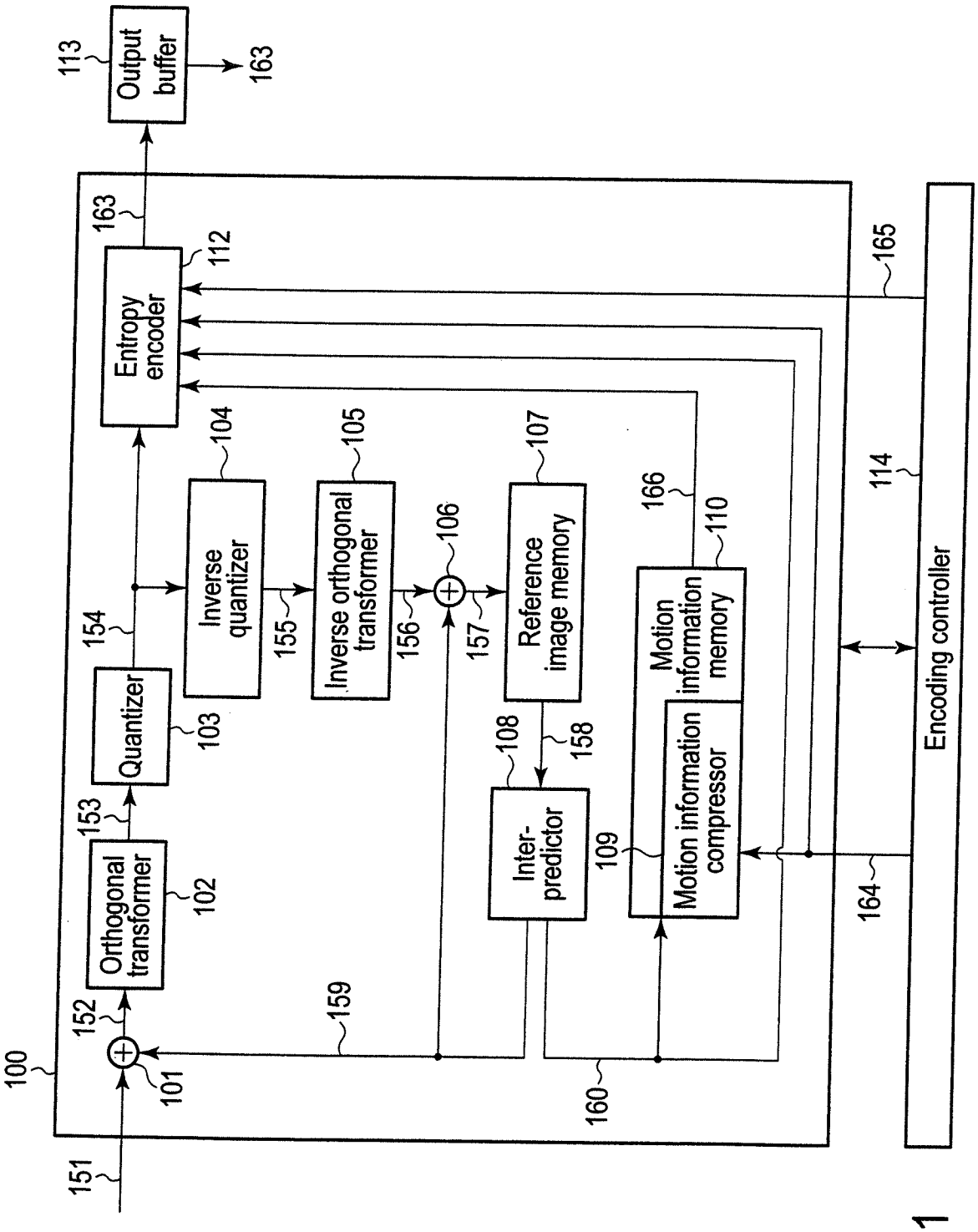

100
101
102
103
104
105
106
107
108
109
110
112
113
114
151
152
153
154
155
156
157
158
159
160
163
164
165
166
Output buffer
Entropy encoder
Quantizer
Inverse quantizer
Inverse orthogonal transformer
Reference image memory
Motion information memory
Motion information compressor
Inter-predictor
Orthogonal transformer
Encoding controller

FIG. 1

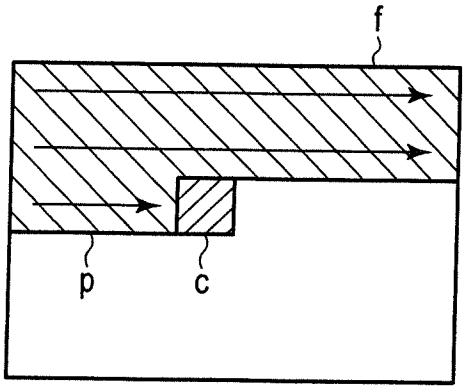


FIG. 2A

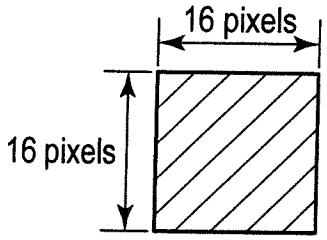


FIG. 2B

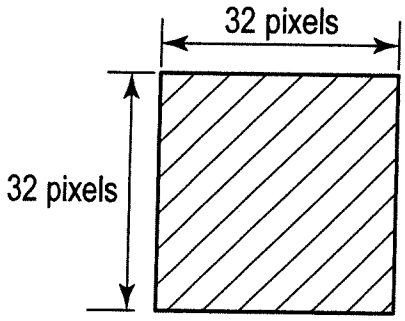


FIG. 2C

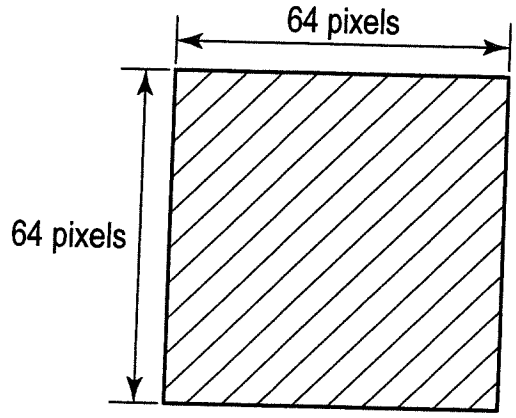


FIG. 2D

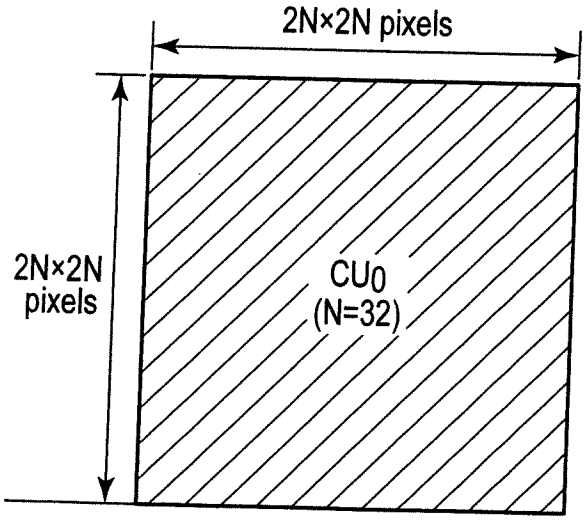


FIG. 3A

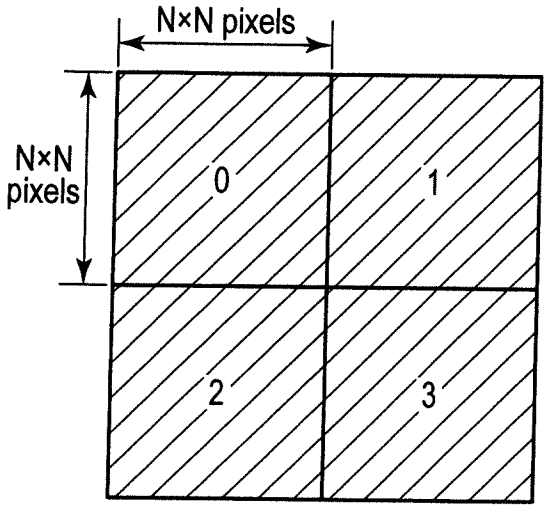


FIG. 3B

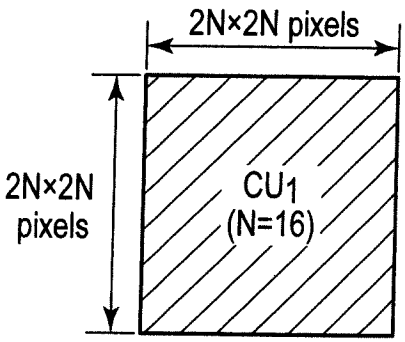


FIG. 3C

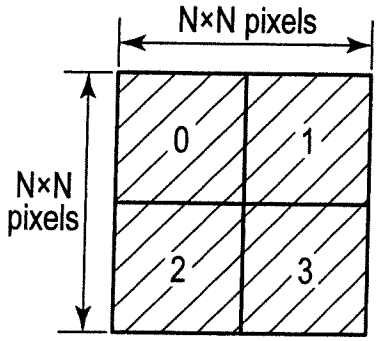


FIG. 3D

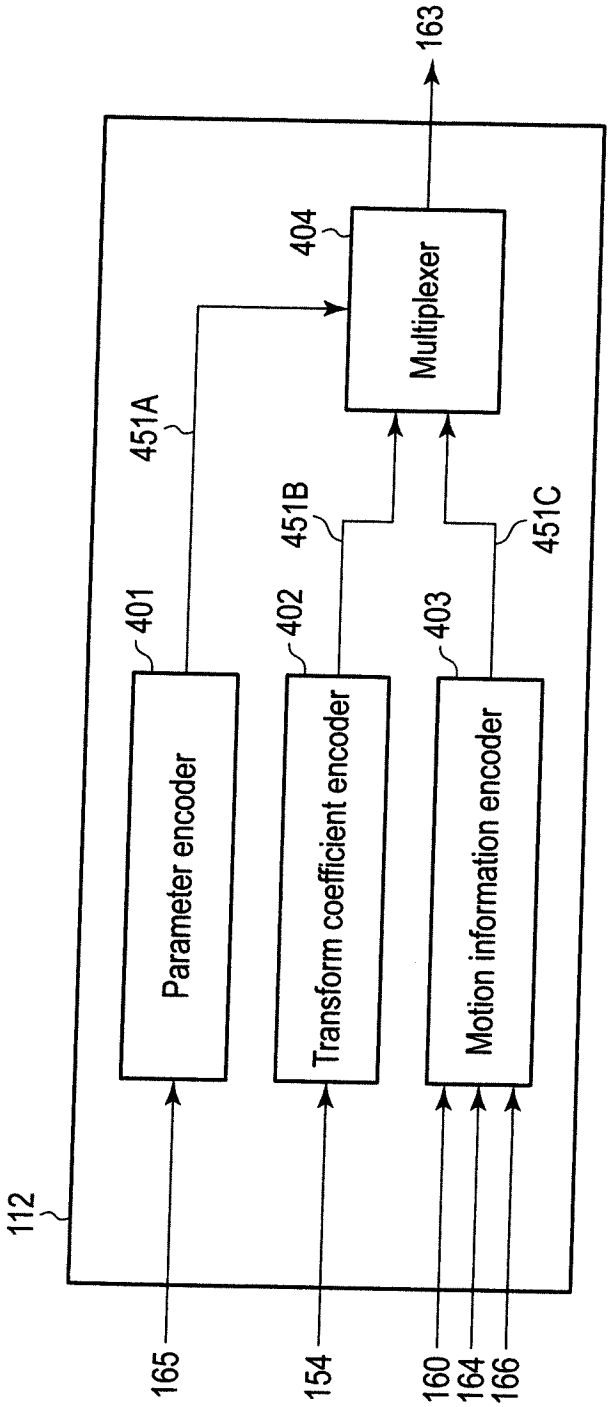

163
404
Multiplexer
451A
451B
451C
401
402
403
Parameter encoder
Transform coefficient encoder
Motion information encoder
112
165
154
160
164
166

FIG. 4

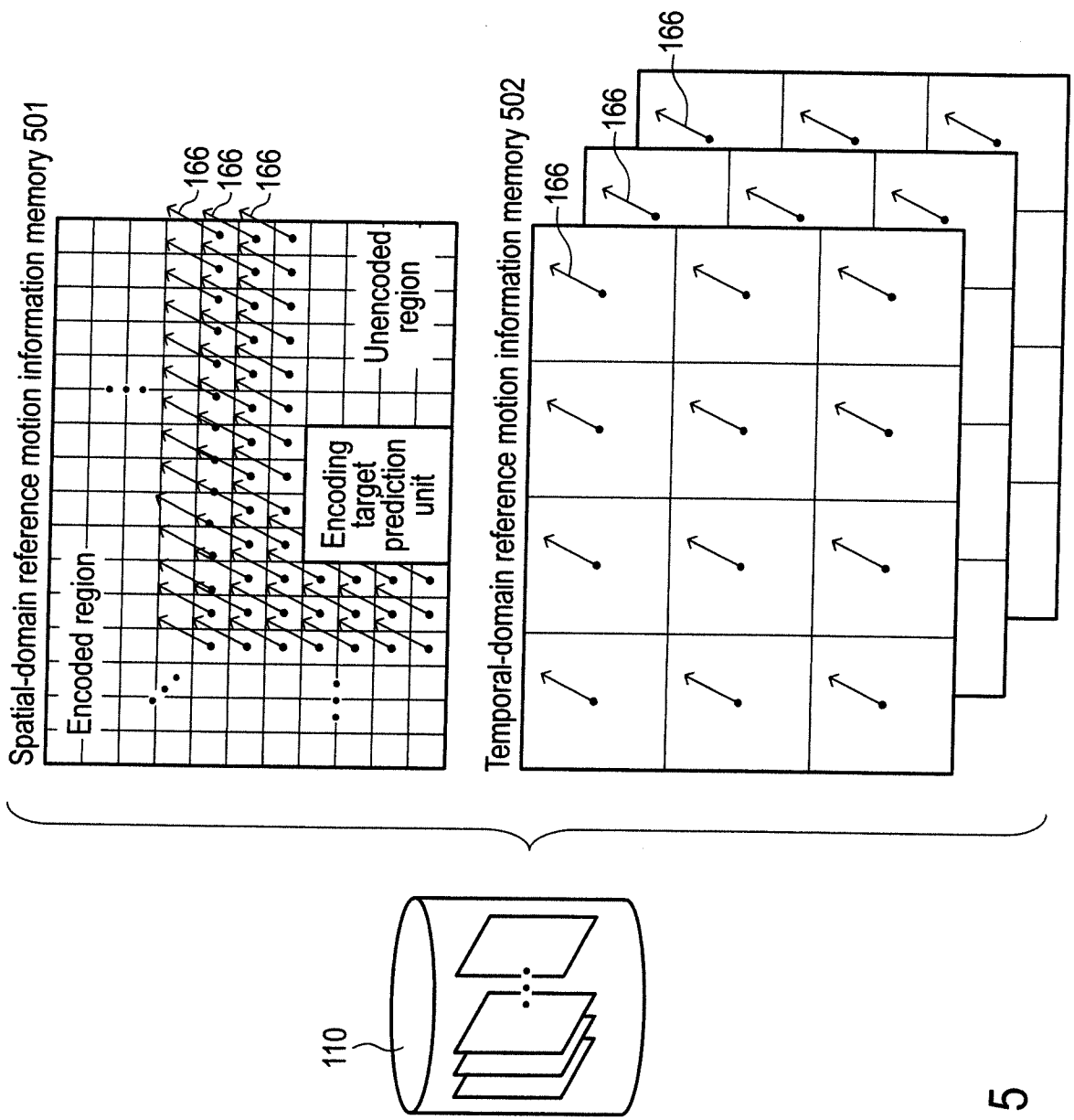

Spatial-domain reference motion information memory 501
Encoded region
166
166
166
Encoding target prediction unit
Unencoded region
Temporal-domain reference motion information memory 502
166
166
166
110

FIG. 5

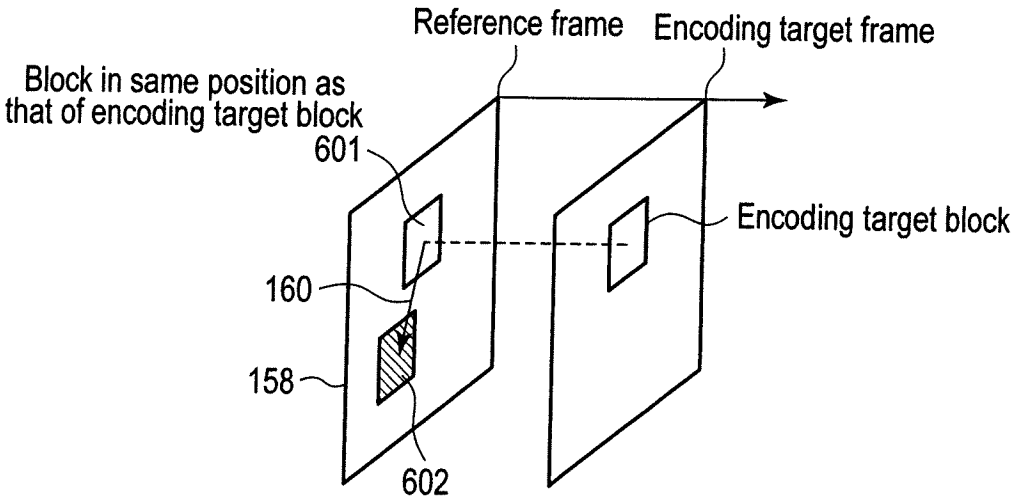

Reference frame
Encoding target frame
Block in same position as that of encoding target block
601
Encoding target block
160
158
602

FIG. 6A

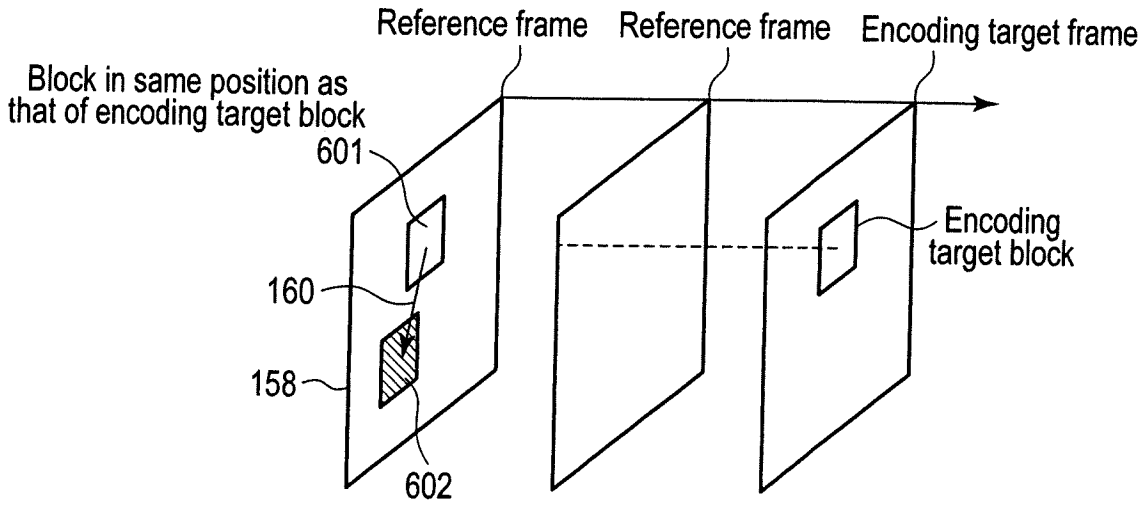

Reference frame
Reference frame
Encoding target frame
Block in same position as that of encoding target block
601
Encoding target block
160
158
602

FIG. 6B

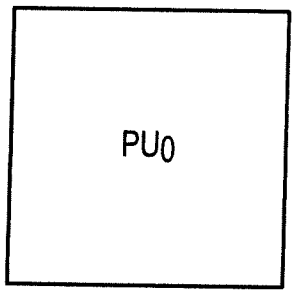


FIG. 7A

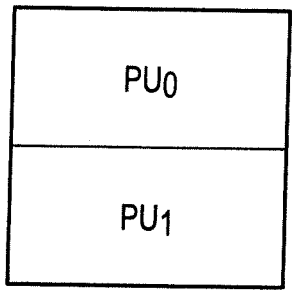


FIG. 7B

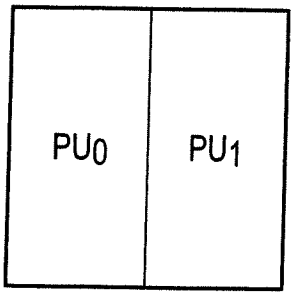


FIG. 7C

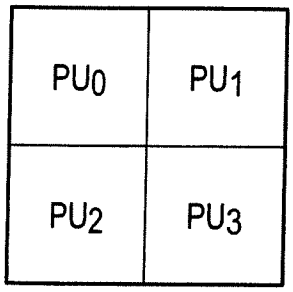


FIG. 7D

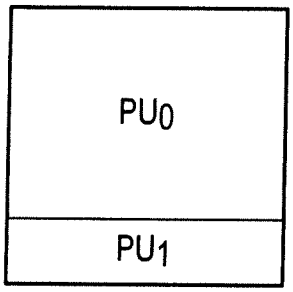


FIG. 7E

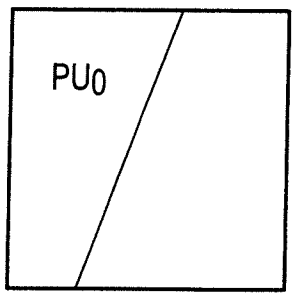


FIG. 7F

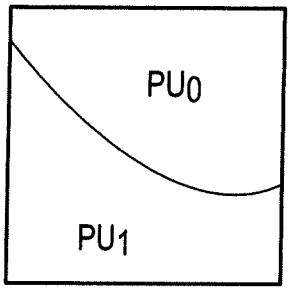


FIG. 7G

	Predicted motion information position encoding	Differential motion information encoding	Transform coefficient encoding
Skip mode	Included	Not included	Not included
Merge mode	Included	Not included	Included
Inter mode	Included	Included	Included

FIG. 8

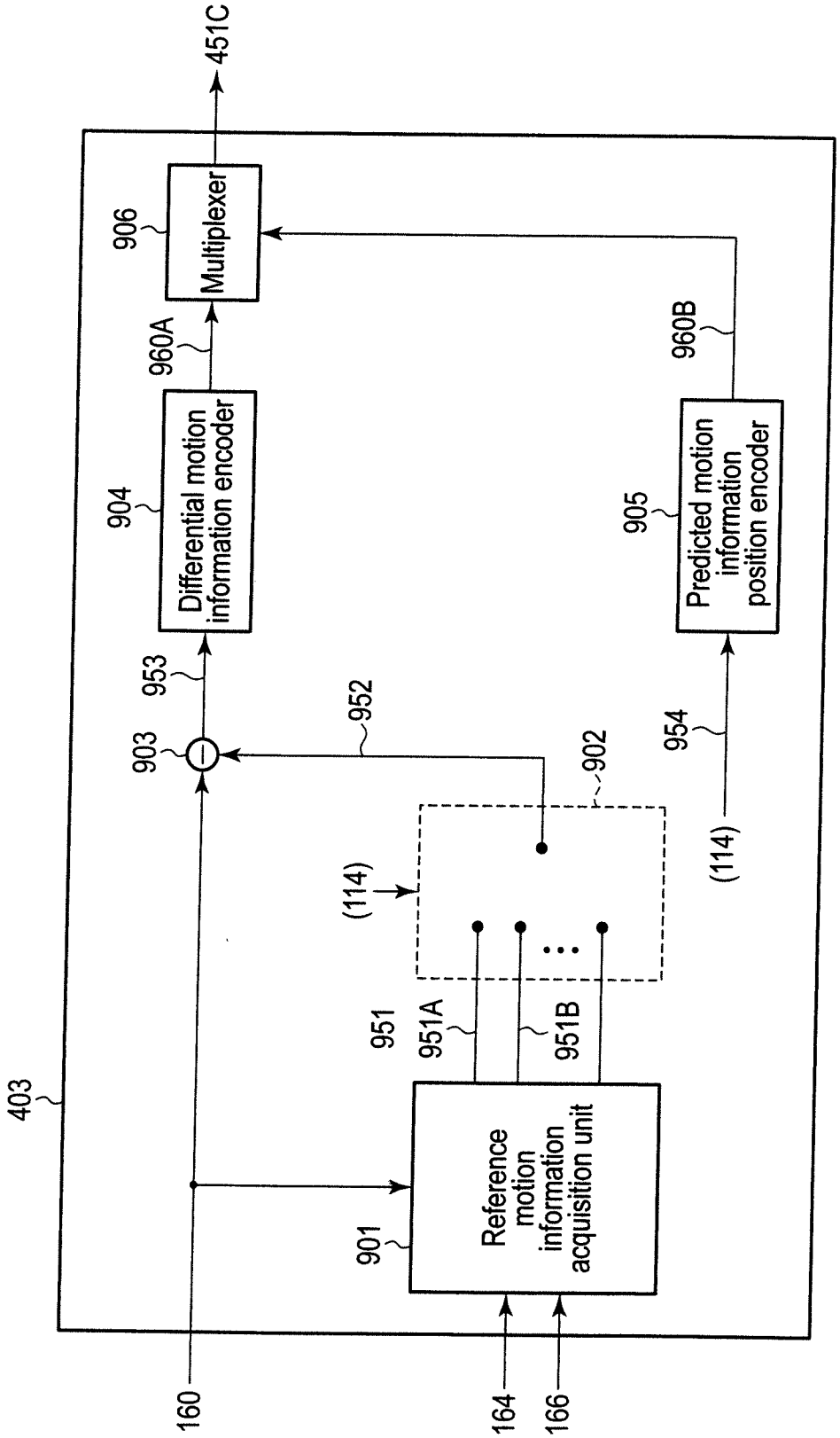

451C
906
Multiplexer
960A
960B
904
Differential motion information encoder
905
Predicted motion information position encoder
953
952
954
903
902
(114)
(114)
951
951A
951B
403
901
Reference motion information acquisition unit
160
164
166

FIG. 9

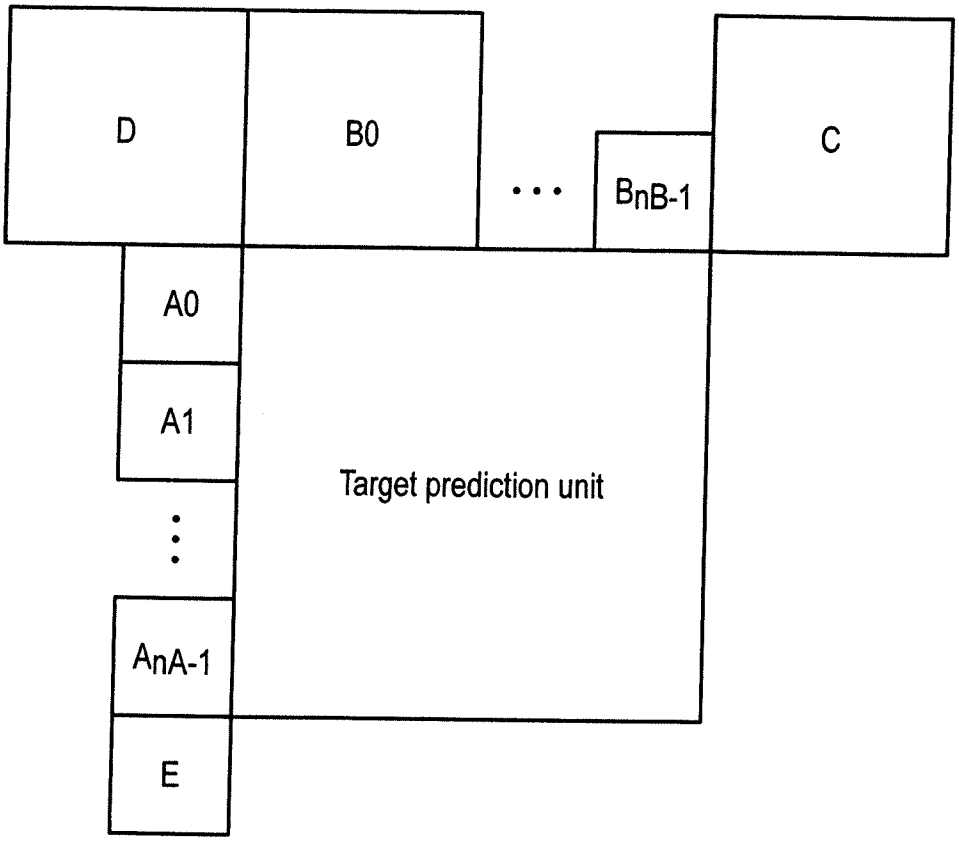

D
B0
BnB-1
C
A0
A1
Target prediction unit
AnA-1
E

FIG. 10

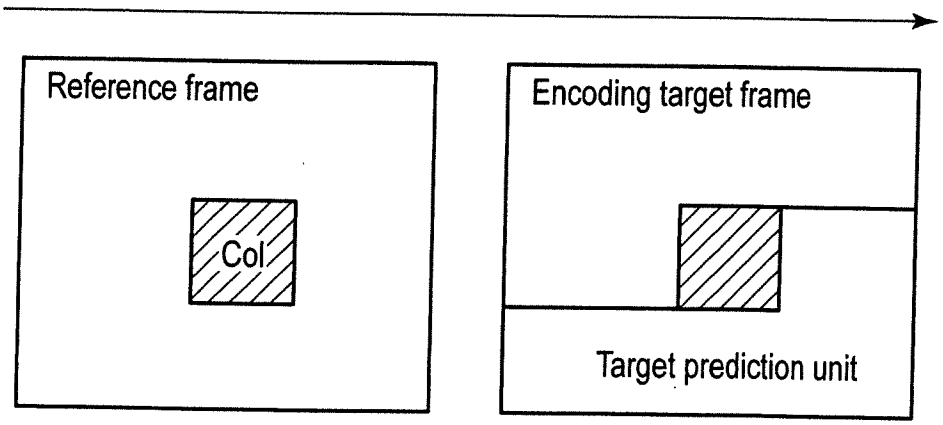

Reference frame
Col
Encoding target frame
Target prediction unit

FIG. 11

Mvpidx	Block position
0	A
1	B
2	C
3	Col

FIG. 12

32x32

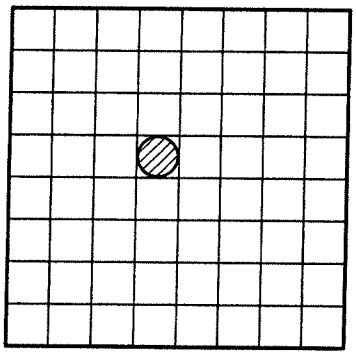


FIG. 13A

32x16

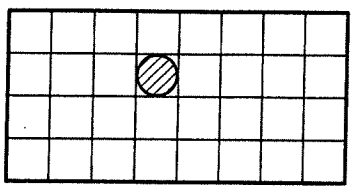


FIG. 13B

16x32

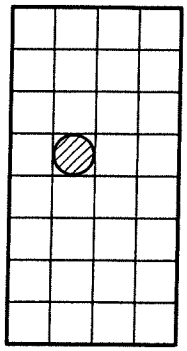


FIG. 13C

16x16

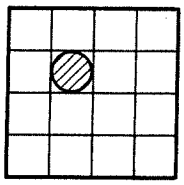


FIG. 13D

16x8

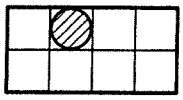


FIG. 13E

8x16

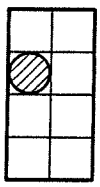


FIG. 13F

32x32

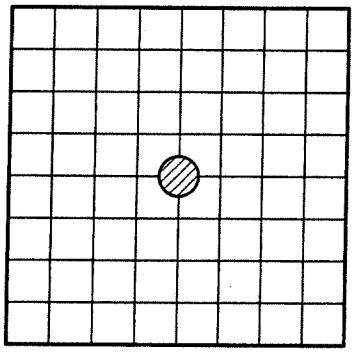


FIG. 14A

32x16

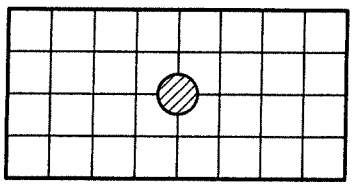


FIG. 14B

16x32

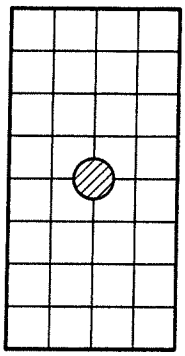


FIG. 14C

16x16

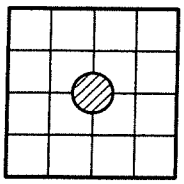


FIG. 14D

16x8

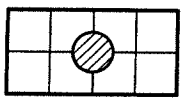


FIG. 14E

8x16

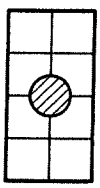


FIG. 14F

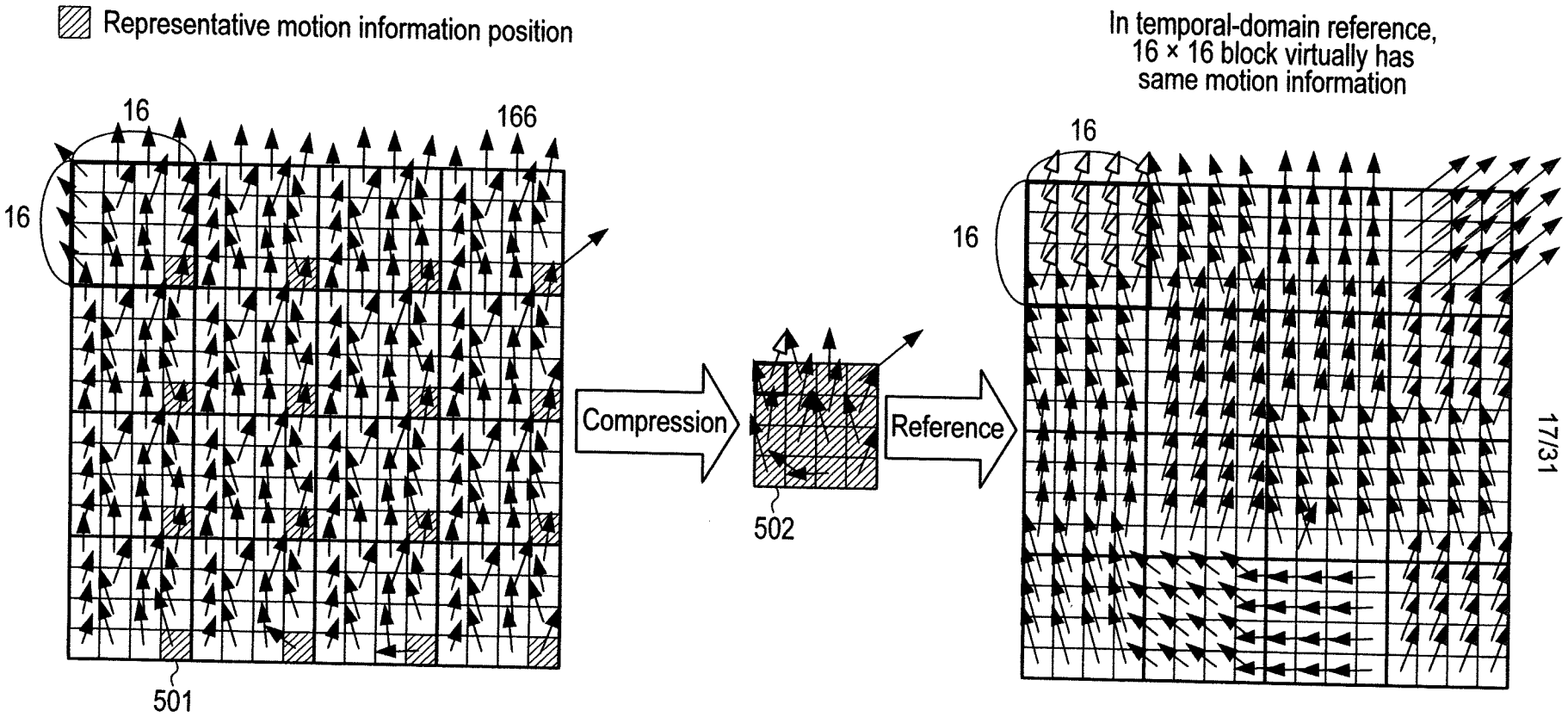

Representative motion information position
16
166
16
501
Compression
502
Reference
In temporal-domain reference,
16 × 16 block virtually has
same motion information
16
16

FIG. 15

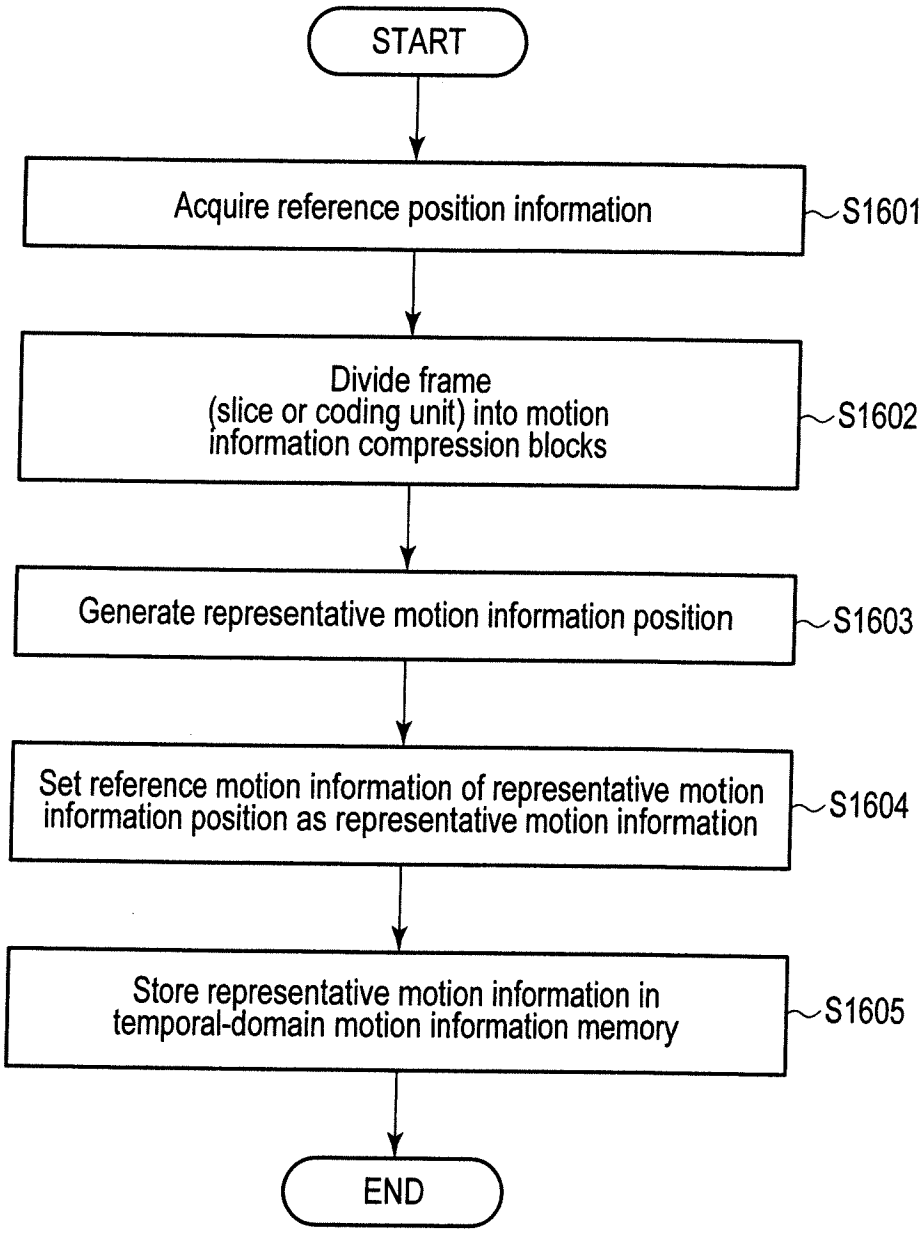

START
Acquire reference position information
S1601
Divide frame (slice or coding unit) into motion information compression blocks
S1602
Generate representative motion information position
S1603
Set reference motion information of representative motion information position as representative motion information
S1604
Store representative motion information in temporal-domain motion information memory
S1605
END

FIG. 16

32x32

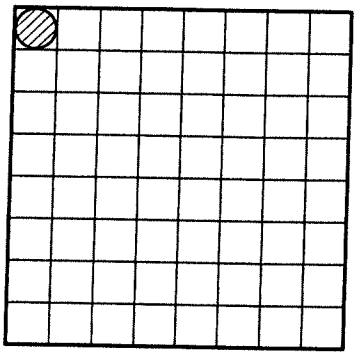


FIG. 17A

32x16

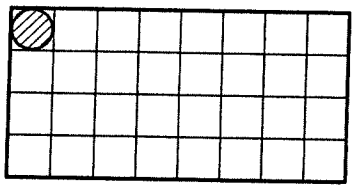


FIG. 17B

16x32

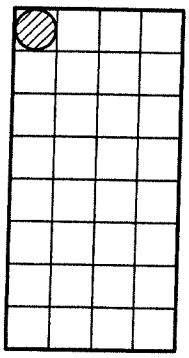


FIG. 17C

16x16

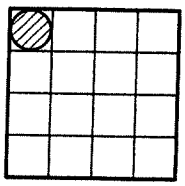


FIG. 17D

16x8

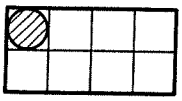


FIG. 17E

8x16

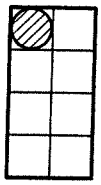


FIG. 17F

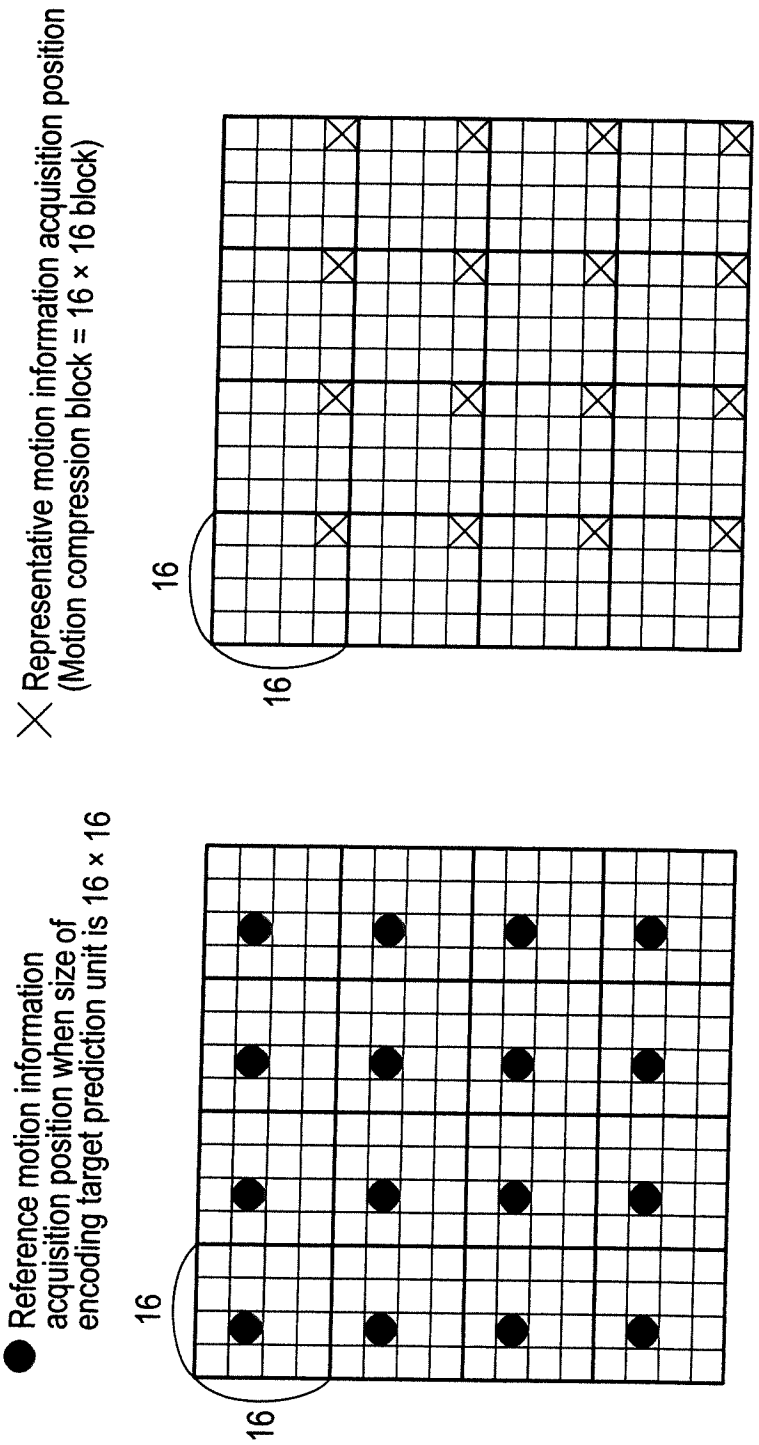

● Reference motion information acquisition position when size of encoding target prediction unit is 16 × 16
16
16
× Representative motion information acquisition position (Motion compression block = 16 × 16 block)
16
16

FIG. 18A

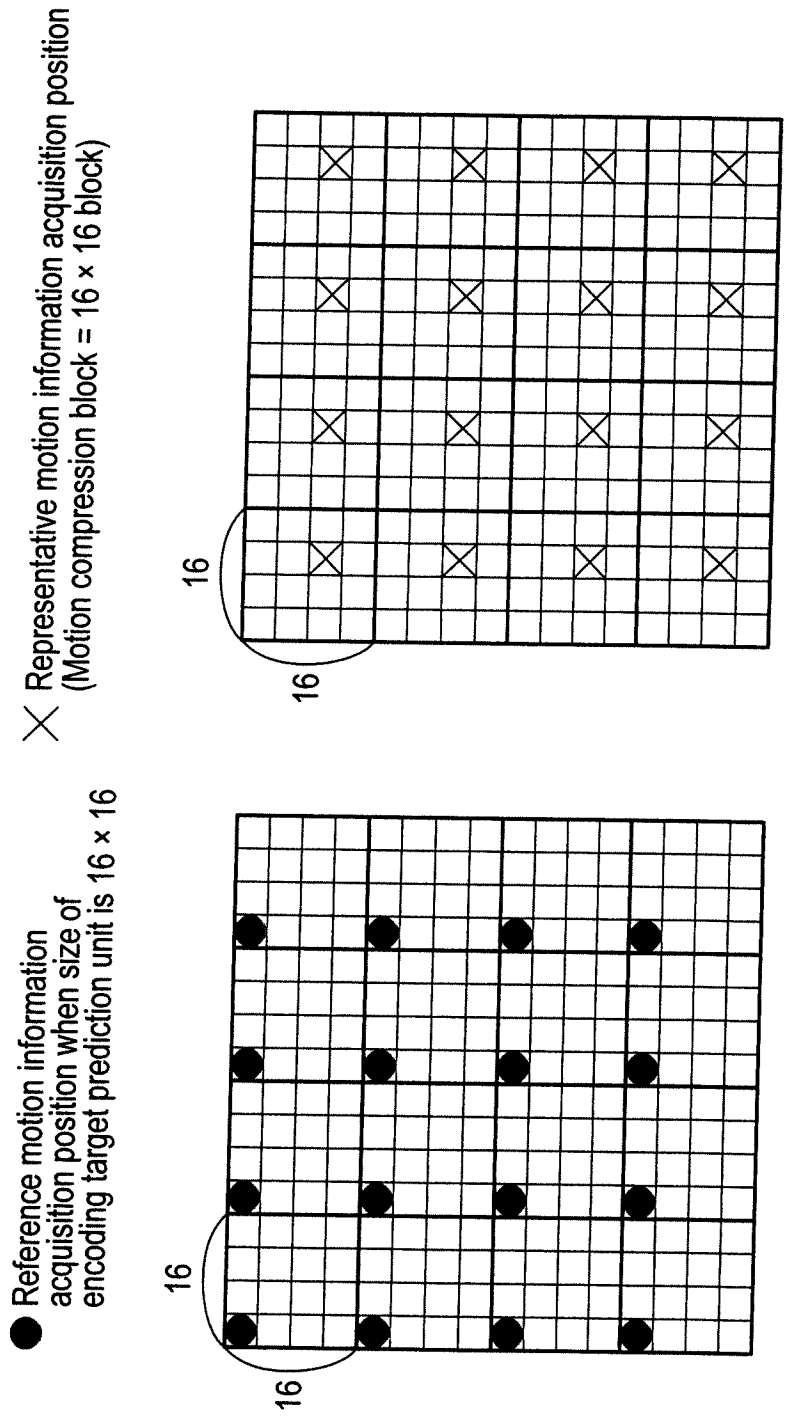

● Reference motion information acquisition position when size of encoding target prediction unit is 16 × 16
× Representative motion information acquisition position (Motion compression block = 16 × 16 block)
16
16
16
16

FIG. 18B

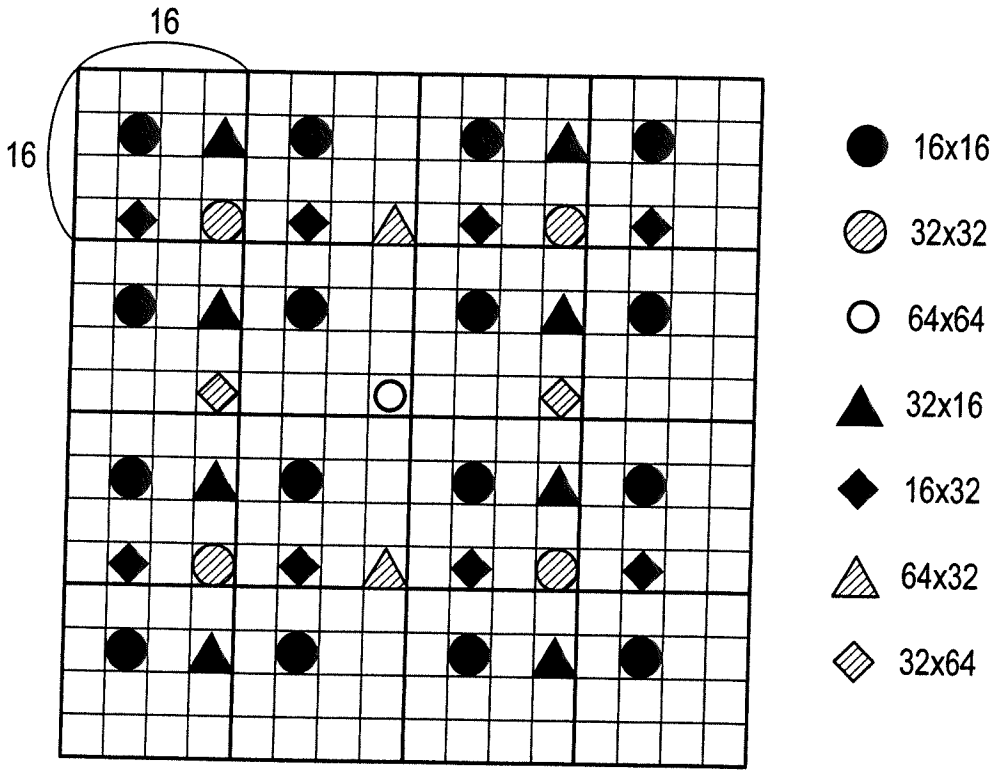


FIG. 19

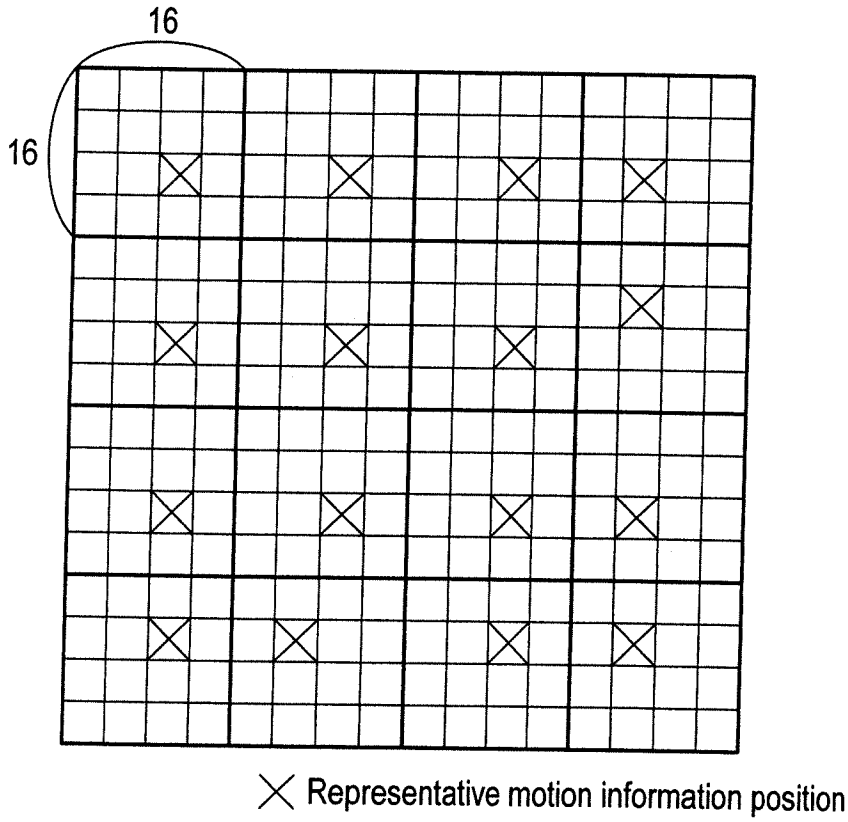


FIG. 20A

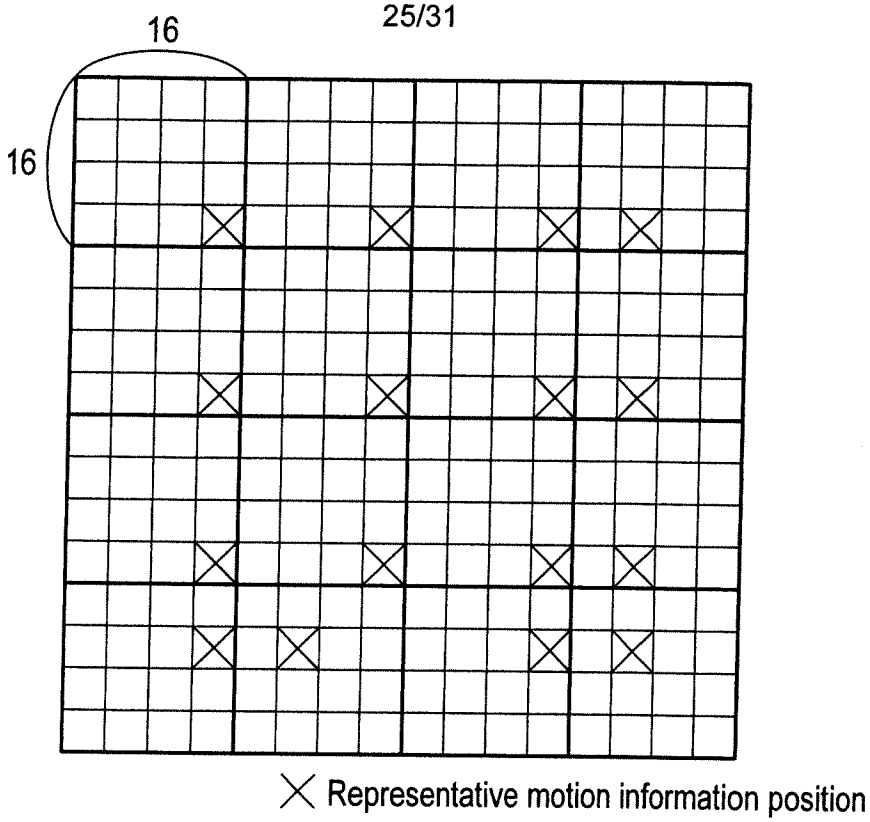


FIG. 20B

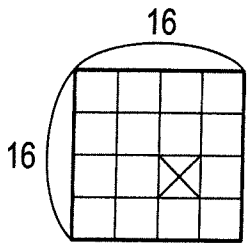


FIG. 21A

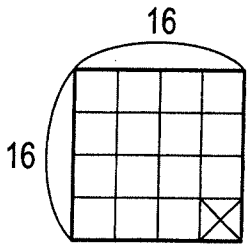


FIG. 21B

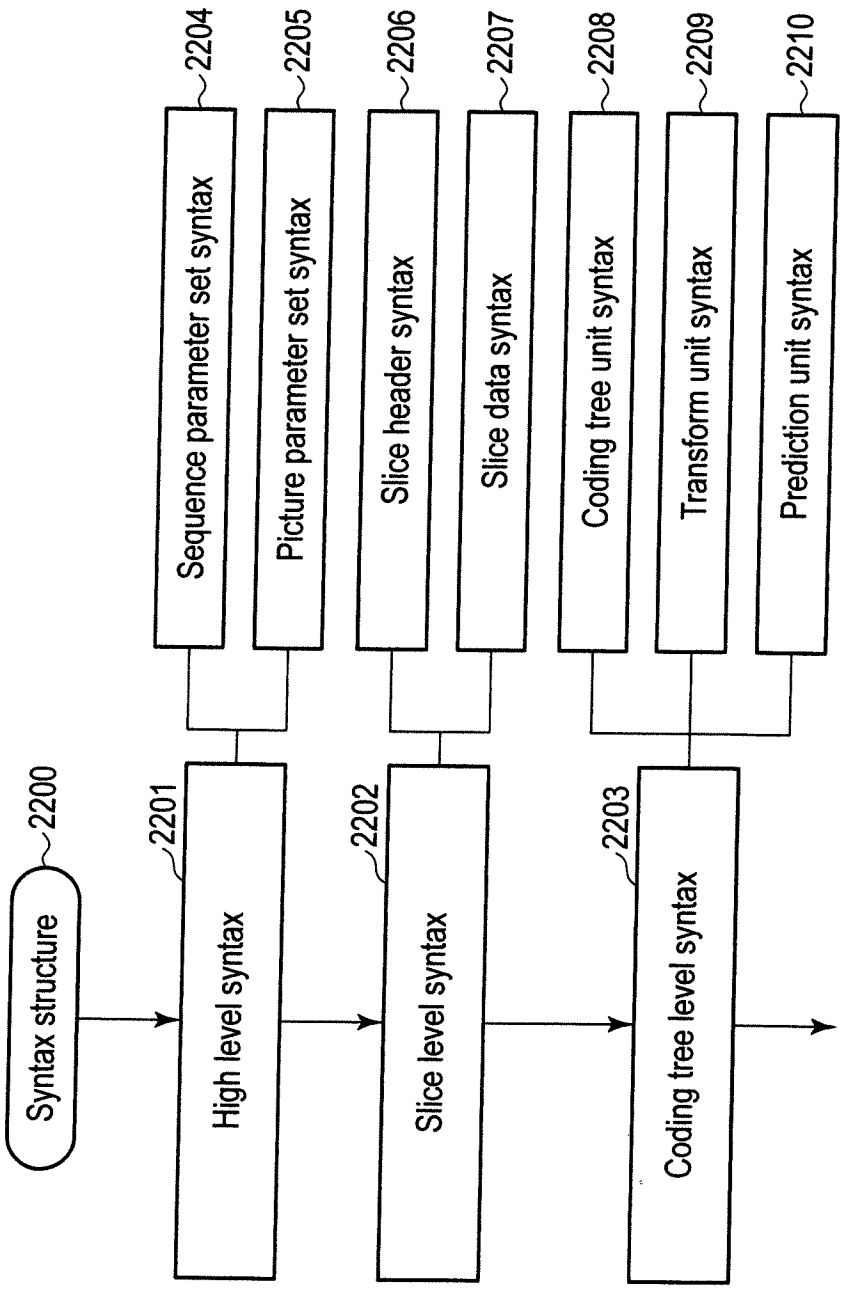

Syntax structure
2200
High level syntax
2201
Slice level syntax
2202
Coding tree level syntax
2203
Sequence parameter set syntax
2204
Picture parameter set syntax
2205
Slice header syntax
2206
Slice data syntax
2207
Coding tree unit syntax
2208
Transform unit syntax
2209
Prediction unit syntax
2210

FIG. 22

```
seq parameter set rbsp( ) {
    .....
    motion_vector_buffer_comp_flag
    if ( motion_vector_buffer_comp_flag )
        motion_vector_buffer_comp_ratio_log2
    rbsp trailing bits( )
}
```

FIG. 23A

```
seq parameter set rbsp( ) {
    .....
    motion_vector_buffer_comp_flag
    if ( motion_vector_buffer_comp_flag ){
        motion_vector_buffer_comp_ratio_log2
        motion_vector_buffer_comp_position
    }
    rbsp trailing bits( )
}
```

FIG. 23B

```
prediction unit( x0, y0, log2PUWidth, log2PUHeight ) {
    if( skip flag[ x0 ][ y0 ] {
        if( NumMVPCand( L0 ) > 1)
            mvp idx 10[ x0 ][ y0 ]
        if( NumMVPCand( L1 ) > 1)
            mvp idx 11[ x0 ][ y0 ]
    } else if( PredMode == MODE INTRA ) {
        /* MODE INTRA*/
        ...
    } else {/* MODE MERGE, MODE INTER*/
        if( NumMergeCandidates > 0)
            merge flag[ x0 ][ y0 ]
        if( merge flag[ x0 ][ y0 ] && NumMergeCandidates > 1)
            merge idx[ x0 ][ y0 ]
        } else {
            /* MODE INTER*/
            if( slice type == B)
                inter pred idc[ x0 ][ y0 ]
            if( inter pred idc[ x0 ][ y0 ] ! = pred L1 {
                if( num ref idx 10 active minus1 > 0 )
                    ref idx 10[ x0 ][ y0 ]
                mvd 10[ x0 ][ y0 ][ 0 ]
                mvd 10[ x0 ][ y0 ][ 1 ]
                if( NumMVPCand( L0 ) > 1)
                    mvd 10[ x0 ][ y0 ]
            }
            if( inter pred idc[ x0 ][ y0 ] ! = pred  L0) {
                if( num ref idx 11 active minus1 > 0 )
                    ref idx 11[ x0 ][ y0 ]
                mvd 11[ x0 ][ y0 ][ 0 ]
                mvd 11[ x0 ][ y0 ][ 1 ]
                if( NumMVPCand( L1 ) > 1)
                    mvd 11[ x0 ][ y0 ]
            }
        }
    }
}
```

FIG. 24

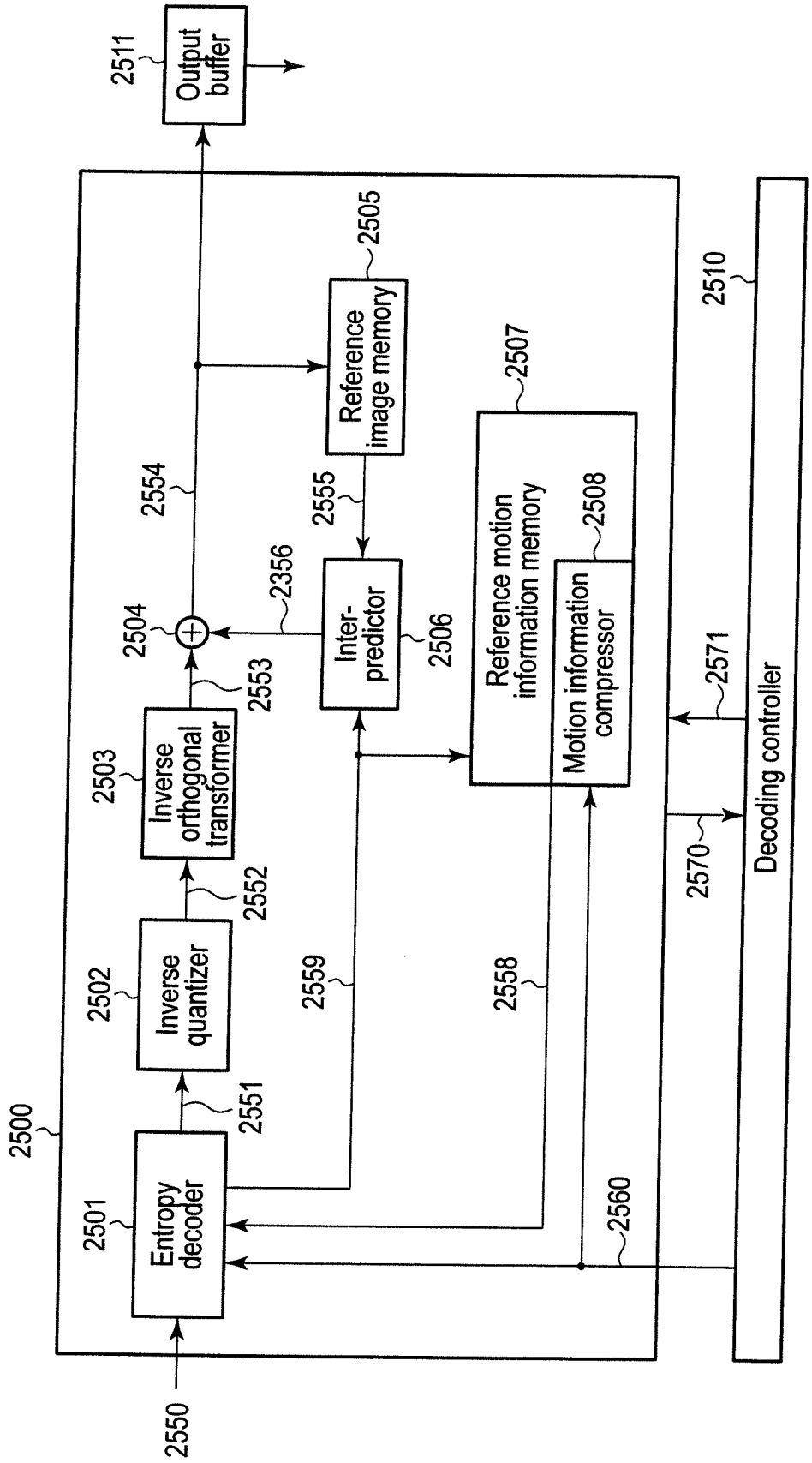

2550
2500
2501
Entropy decoder
2551
2502
Inverse quantizer
2552
2503
Inverse orthogonal transformer
2553
2504
2554
2505
Reference image memory
2555
2356
Inter-predictor
2506
2507
Reference motion information memory
2508
Motion information compressor
2559
2558
2560
2570
2571
2510
Decoding controller
2511
Output buffer

FIG. 25

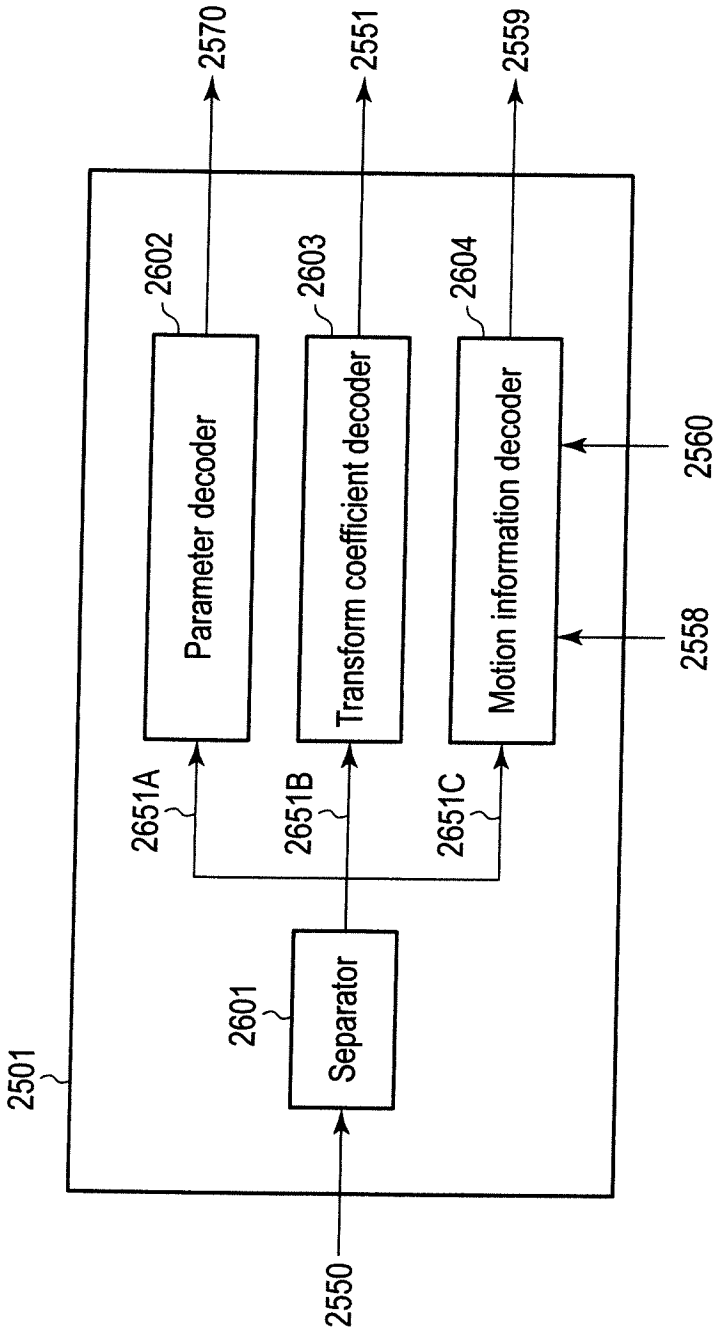

2570
2551
2559
2602
2603
2604
Parameter decoder
Transform coefficient decoder
Motion information decoder
2560
2558
2651A
2651B
2651C
2601
Separator
2501
2550

FIG. 26

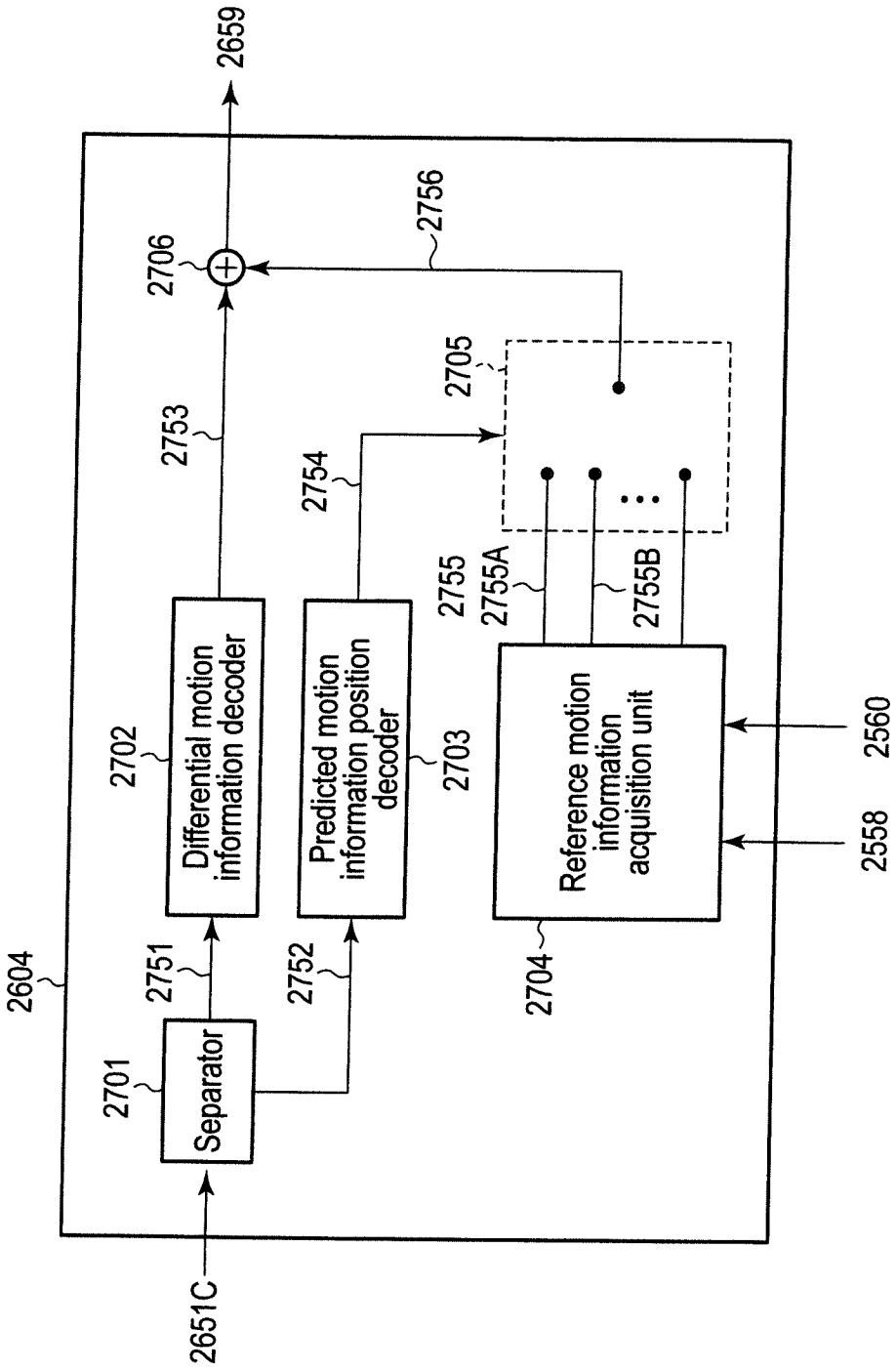

2659
2706
2756
2705
2753
2754
2755
2755A
2755B
2702
2703
2704
2604
2751
2752
2701
Separator
Differential motion information decoder
Predicted motion information position decoder
Reference motion information acquisition unit
2560
2558
2651C

FIG. 27