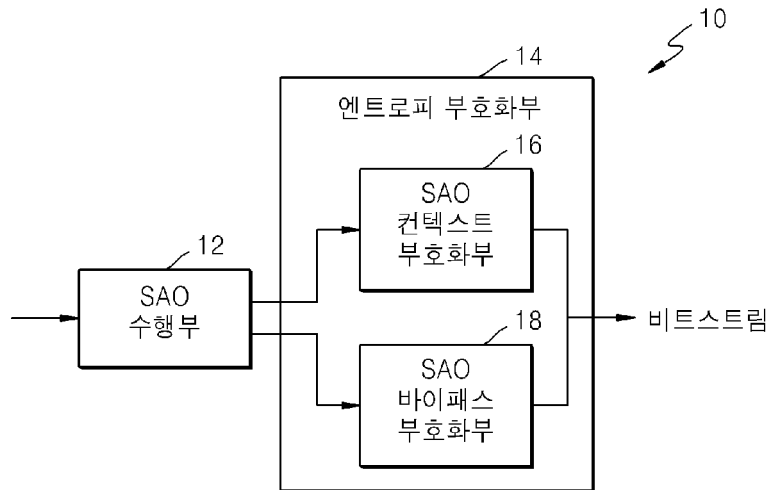
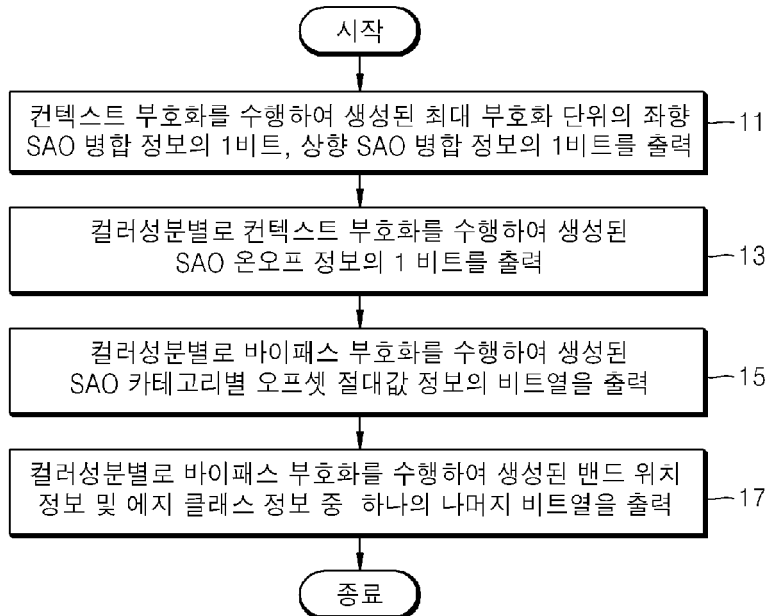


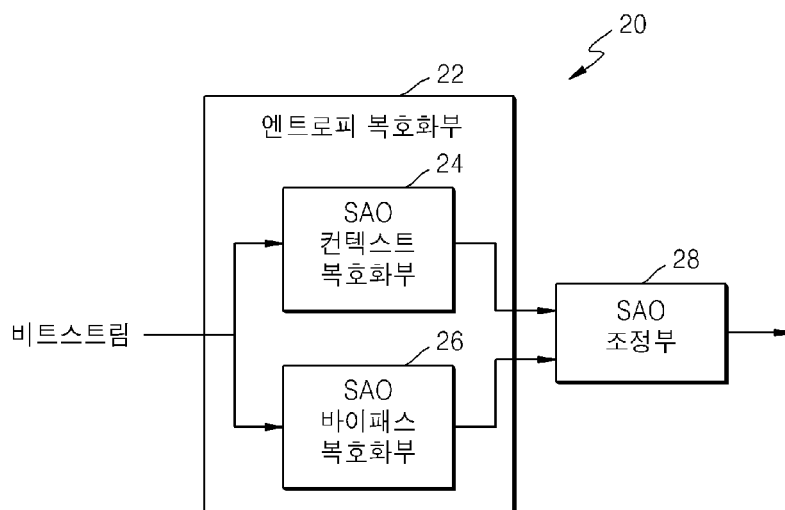
[Fig. 1a]



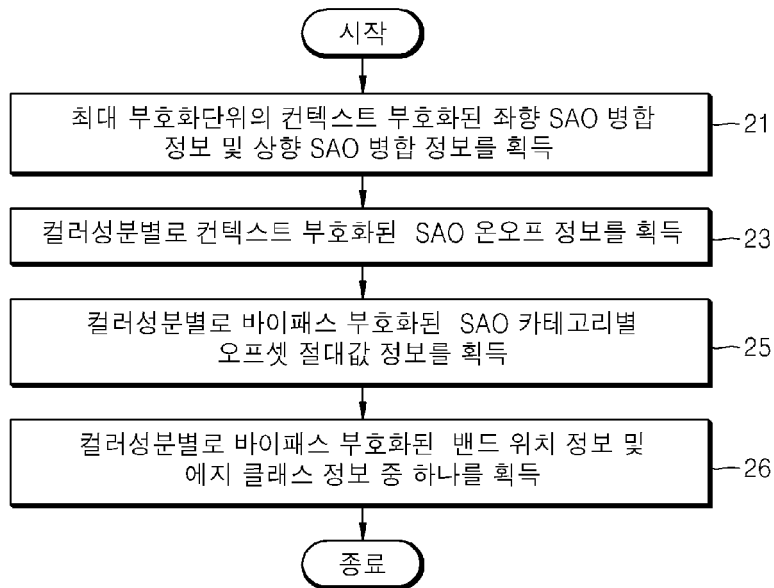
[Fig. 1b]



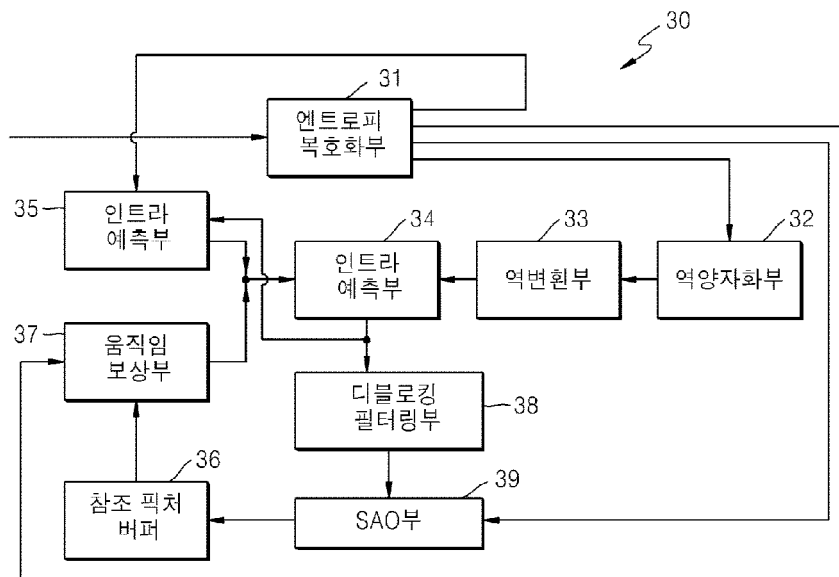
[Fig. 2a]



[Fig. 2b]



[Fig. 3]



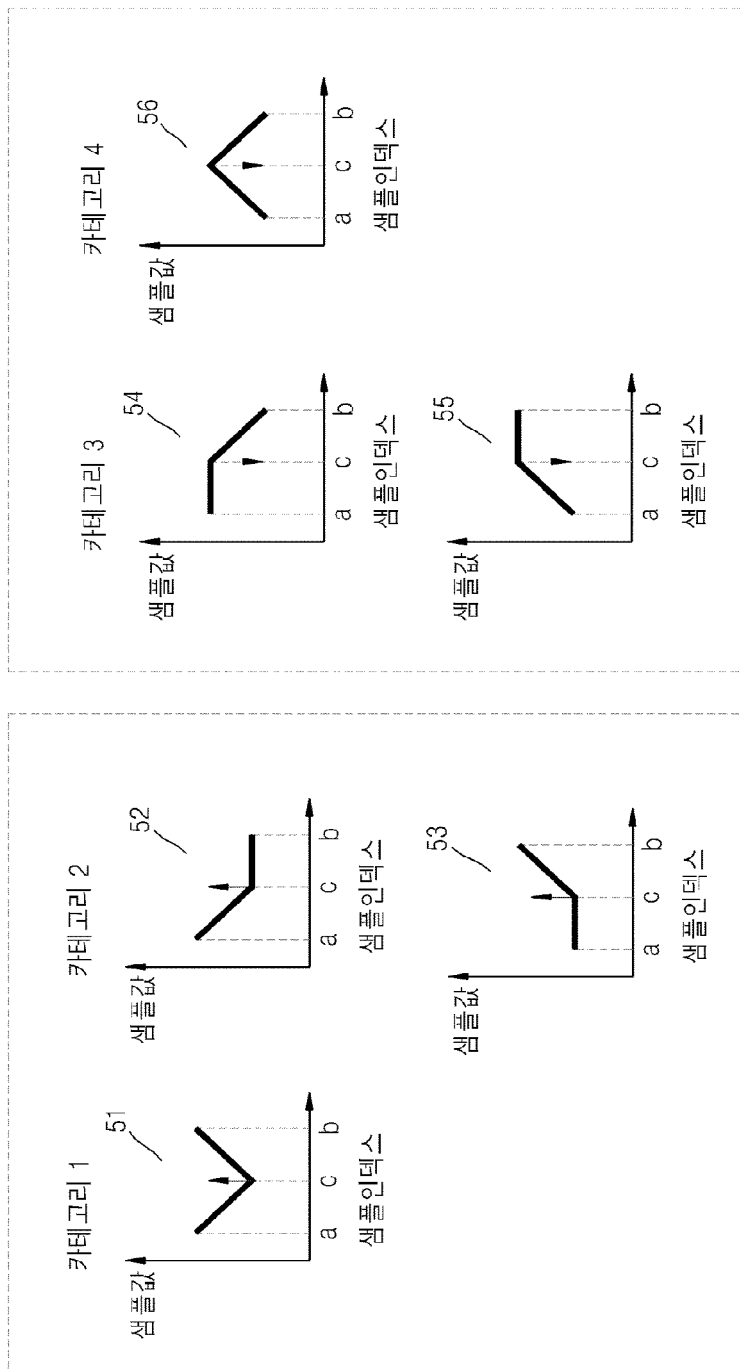
[Fig. 4]

41	42	43	44
EO 클래스 = 0	EO 클래스 = 1	EO 클래스 = 2	EO 클래스 = 3
수평	수직	135° 대각선	45° 대각선
X1 X0 X2	X3 X0 X4	X5 X0 X8	X6 X0 X7

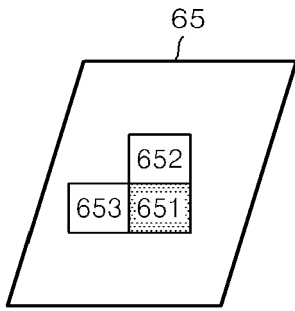
[Fig. 5a]

카테고리	조건
1	$X_c < X_a \ \&\& \ X_c < X_b$
2	$(X_c < X_a \ \&\& \ X_c == X_b) \parallel (X_c == X_a \ \&\& \ X_c < X_b)$
3	$(X_c > X_a \ \&\& \ X_c == X_b) \parallel (X_c == X_a \ \&\& \ X_c > X_b)$
4	$X_c > X_a \ \&\& \ X_c > X_b$
0	카테고리 1, 2, 3, 4 조건 불충족시

[Fig. 5b]



[Fig. 6a]



[Fig. 6b]

60

```

for( cldx = 0; cldx < 3; cldx++ ){ // 3종 컬러 성분들에 대해 루프 동작
    sao_merge_left_flag           // 컨텍스트 부호화된 1개 bin
    sao_merge_up_flag            // 컨텍스트 부호화된 1개 bin
    sao_type_idx                  // 컨텍스트 부호화된 6개 bin
    sao_band_position             // 바이패스 부호화된 5개 bin
        for( i = 0; i < 4; i++ ){ // 4종 SAO 카테고리들에 대해 루프 동작
            sao_offset_abs         // 컨텍스트 부호화된 31개 bin
            sao_offset_sign        // 바이패스 부호화된 1개 bin
        }
    }
}

```

[Fig. 7a]

```

70
sao_param( rx, ry){
  if( rx > 0 ) {
    leftCtblnSlice = CtbAddrInSlice > 0
    leftCtblnTile = TileId[ CtbAddrTS ] == TileId[ CtbAddrRStoTS[ CtbAddrRS - 1 ] ]
    if( leftCtblnSlice && leftCtblnTile )
71 ——— sao_merge_left_flag
    }
    if( ry > 0 && !sao_merge_left_flag ) {
      upCtblnSlice = ( CtbAddrTS - CtbAddrRStoTS[ CtbAddrRS - PicWidthInCtbs ] ) <= CtbAddrInSlice
      upCtblnTile = TileId[ CtbAddrTS ] == TileId[ CtbAddrRStoTS[ CtbAddrRS - PicWidthInCtbs ] ]
      if( upCtblnSlice && upCtblnTile )
72 ——— sao_merge_up_flag
    }
    if( !sao_merge_up_flag && !sao_merge_left_flag ) {
73 ——— for( cldx = 0; cldx < 3; cldx++){
      if( ( slice_sao_flag_luma && cldx == 0 ) || ( slice_sao_flag_chroma && cldx > 0 ) ) {
74 ——— if ( cldx == 0 ){
        741 ——— sao_on_off_flag_luma[ rx ][ ry ]
        if ( sao_on_off_flag_luma[ rx ][ ry ] )
        743 ——— sao_eo_bo_flag_luma[ rx ][ ry ]
        }
        75 ——— if ( cldx == 1 ){
          751 ——— sao_on_off_flag_chroma[ rx ][ ry ]
          if ( sao_on_off_flag_chroma[ rx ][ ry ] )
          753 ——— sao_eo_bo_flag_chroma[ rx ][ ry ]
          }
          76 ——— if ( SaoTypeIdx[cldx][ rx ][ ry ] != 0 ) {
            for( i = 0; i < 4; i++ )
            77 ——— sao_offset_abs[ cldx ][ rx ][ ry ][ i ]
            78 ——— if( SaoTypeIdx[ cldx ][ rx ][ ry ] == 1 ) {
              for( i = 0; i < 4; i++ ) {
                if( sao_offset_abs[ cldx ][ rx ][ ry ][ i ] != 0 )
                781 ——— sao_offset_sign[ cldx ][ rx ][ ry ][ i ]
              }
              782 ——— sao_band_position[ cldx ][ rx ][ ry ]
            }
            79 ——— else{
              if ( cldx == 0 ) sao_eo_class_luma[ rx ][ ry ]
              if ( cldx == 1 ) sao_eo_class_chroma[ rx ][ ry ]
              }
            }
          }
        }
      }
    }
  }
}

```

[Fig. 7b]

```

sao_param( rx, ry){
    if( rx > 0 ) {
        leftCtbnSlice = CtbAddrInSlice > 0
        leftCtbnTile = TileId[ CtbAddrTS ] == TileId[ CtbAddrRStoTS[ CtbAddrRS - 1 ] ]
        if( leftCtbnSlice && leftCtbnTile )
71 ——— sao_merge_left_flag
    }
    if( ry > 0 && !sao_merge_left_flag ) {
        upCtbnSlice = ( CtbAddrTS - CtbAddrRStoTS[CtbAddrRS - PicWidthInCtbs]) <= CtbAddrInSlice
        upCtbnTile = TileId[ CtbAddrTS ] == TileId[ CtbAddrRStoTS[CtbAddrRS - PicWidthInCtbs ] ]
        if( upCtbnSlice && upCtbnTile )
72 ——— sao_merge_up_flag
    }
    if( !sao_merge_up_flag && !sao_merge_left_flag ) {
83 ——— sao_on_off_flag_luma[ rx ][ ry ]
        if (sao_on_off_flag_luma[ rx ][ ry ])
831 ——— sao_eo_bo_flag_luma[ rx ][ ry ]
            if (sao_eo_bo_flag_luma[ rx ][ ry ]==1)
833 ——— sao_eo_class_luma[ rx ][ ry ]
84 ——— sao_on_off_flag_chroma[ rx ][ ry ]
            if (sao_on_off_flag_chroma[ rx ][ ry ])
841 ——— sao_eo_bo_flag_chroma[ rx ][ ry ]
                if (sao_eo_bo_flag_chroma[ rx ][ ry ]==1)
843 ——— sao_eo_class_chroma[ rx ][ ry ]
85 ——— for( cldx = 0; cldx < 3; cldx++){
86 ——— if (SaoTypeIdx[cldx][ rx ][ ry ]!=0) {
                for( i = 0; i < 4; i++ )
87 ——— sao_offset_abs[ cldx ][ rx ][ ry ][ i ]
88 ——— if(SaoTypeIdx[cldx][ rx ][ ry ]==1) {
                        for( i = 0; i < 4; i++ ) {
                                if( sao_offset_abs[ cldx ][ rx ][ ry ][ i ] != 0 )
881 ——— sao_offset_sign[ cldx ][ rx ][ ry ][ i ]
                        }
883 ——— sao_band_position[ cldx ][ rx ][ ry ]
                }
            }
        }
    }
}

```

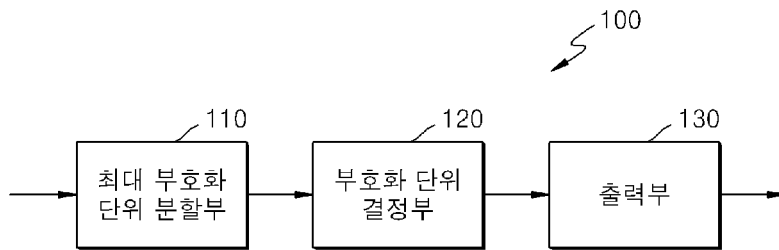
[Fig. 7c]

```

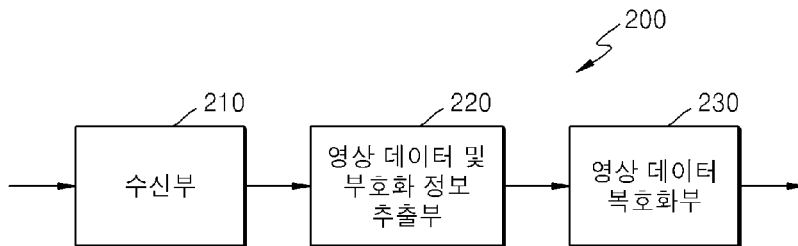
sao_param( rx, ry){
    if( rx > 0 ) {
        leftCtbnSlice = CtbAddrInSlice > 0
        leftCtbnTile = TileId[ CtbAddrTS ] == TileId[ CtbAddrRStoTS[ CtbAddrRS - 1 ] ]
        if( leftCtbnSlice && leftCtbnTile )
71 ——— sao_merge_left_flag
    }
    if( ry > 0 && !sao_merge_left_flag ) {
        upCtbnSlice = ( CtbAddrTS - CtbAddrRStoTS[CtbAddrRS - PicWidthInCtbs]) <= CtbAddrInSlice
        upCtbnTile = TileId[ CtbAddrTS ] == TileId[ CtbAddrRStoTS[CtbAddrRS - PicWidthInCtbs ] ]
        if( upCtbnSlice && upCtbnTile )
72 ——— sao_merge_up_flag
    }
    if( !sao_merge_up_flag && !sao_merge_left_flag ) {
83 ——— sao_on_off_flag_luma[ rx ][ ry ]
84 ——— sao_on_off_flag_chroma[ rx ][ ry ]
        if( sao_on_off_flag_luma[ rx ][ ry ] )
831 ——— sao_eo_bo_flag_luma[ rx ][ ry ]
            if( sao_eo_bo_flag_luma[ rx ][ ry ]==1)
833 ——— sao_eo_class_luma[ rx ][ ry ]
84 ——— sao_on_off_flag_chroma[ rx ][ ry ]
            if( sao_on_off_flag_chroma[ rx ][ ry ] )
841 ——— sao_eo_bo_flag_chroma[ rx ][ ry ]
                if( sao_eo_bo_flag_chroma[ rx ][ ry ]==1)
843 ——— sao_eo_class_chroma[ rx ][ ry ]
85 ——— for( cldx = 0; cldx < 3; cldx++ ){
86 ——— if( SaoTypeIdx[cldx][ rx ][ ry ]!=0) {
                for( i = 0; i < 4; i++ )
87 ——— sao_offset_abs[ cldx ][ rx ][ ry ][ i ]
88 ——— if(SaoTypeIdx[cldx][ rx ][ ry ]==1) {
                        for( i = 0; i < 4; i++ ) {
                                if( sao_offset_abs[ cldx ][ rx ][ ry ][ i ] != 0 )
881 ——— sao_offset_sign[ cldx ][ rx ][ ry ][ i ]
                        }
883 ——— sao_band_position[ cldx ][ rx ][ ry ]
                }
            }
        }
    }
}

```

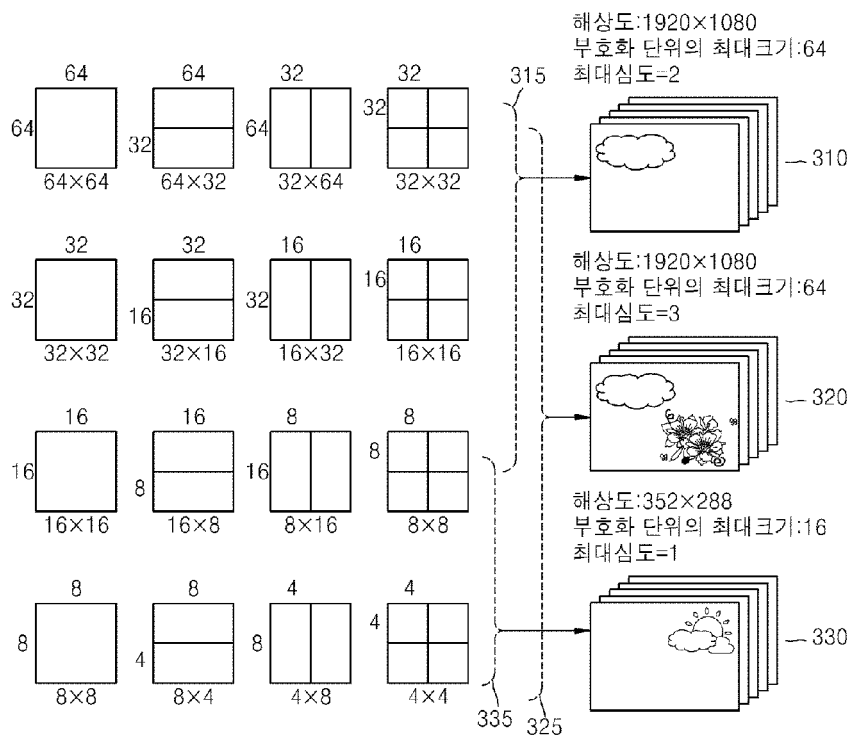
[Fig. 8]



[Fig. 9]



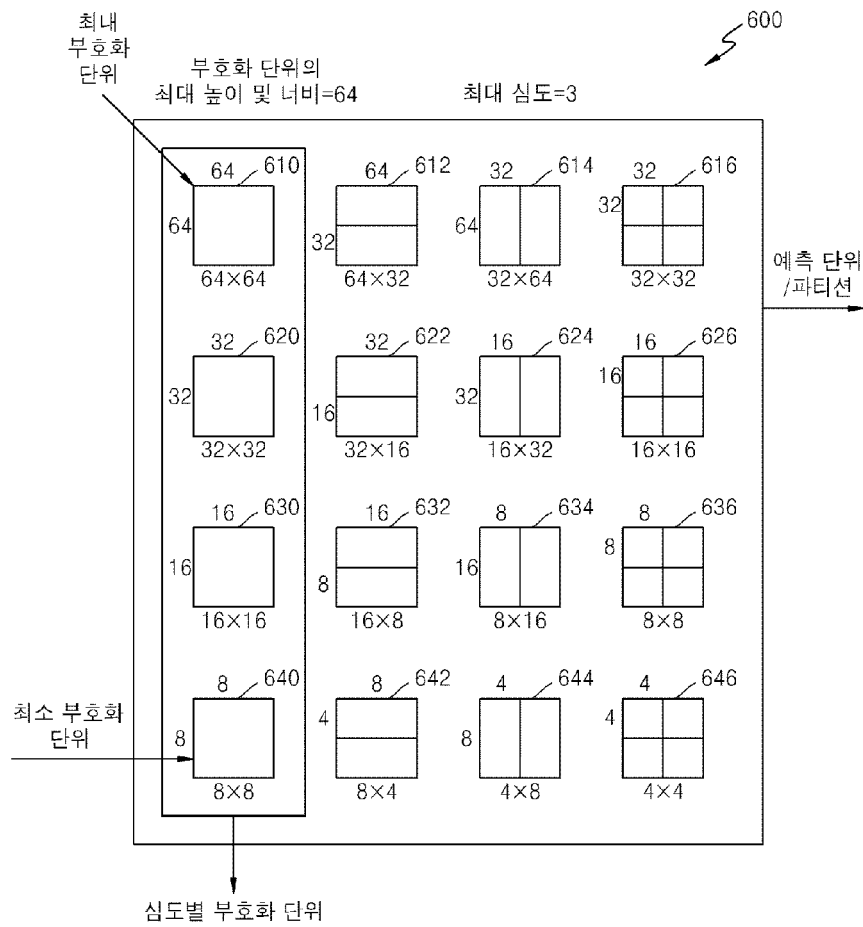
[Fig. 10]



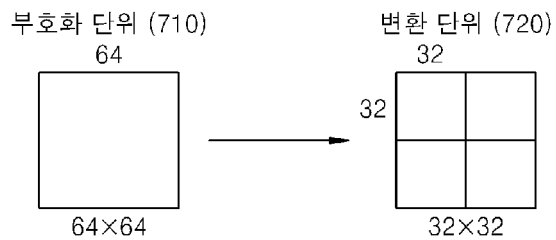
```

graph TD
    405[현재 프레임] --> 410[인트라 예측부]
    495[참조 프레임] --> 420[움직임 추정부]
    410 --> 430[변환부]
    420 --> 430
    420 --> 425[움직임 보상부]
    425 --> 440[양자화부]
    490[오프셋 조정부] --> 430
    430 --> 440[양자화부]
    440 --> 450[엔트로피 부호화부]
    450 --> 455[비트스트림]
    440 --> 460[역양자화부]
    460 --> 470[주파수 역변환부]
    470 --> 480[디블로킹부]
    480 --> 490
    440 --> 420
    450 --> 410
  
```

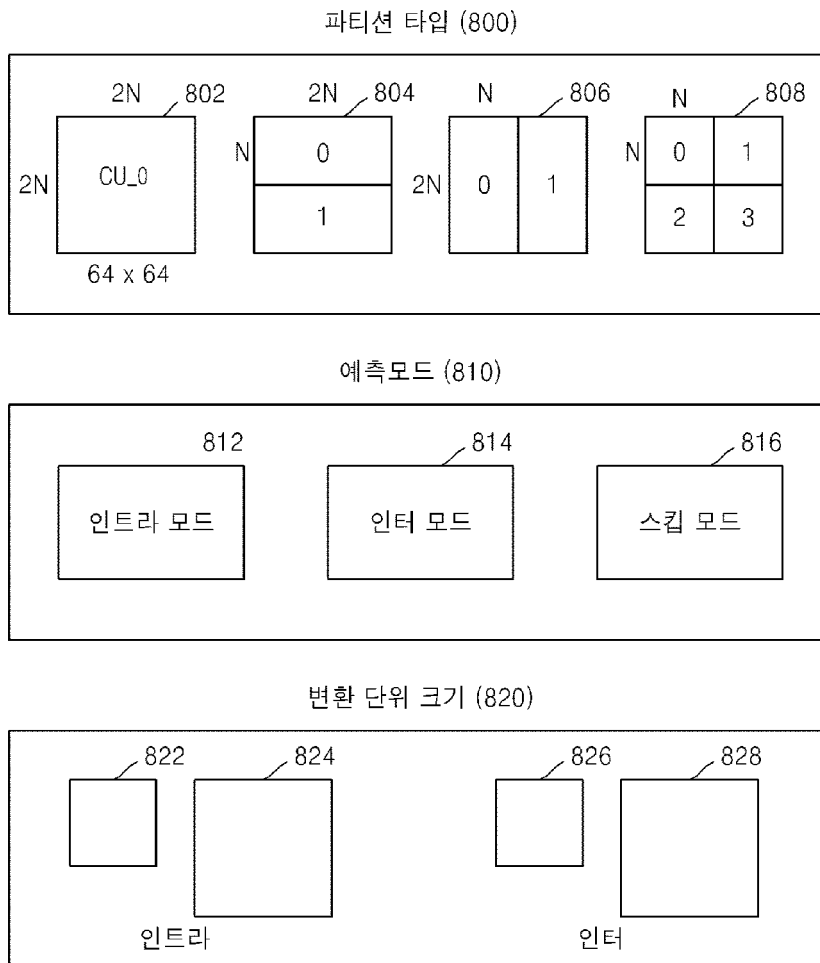
[Fig. 13]



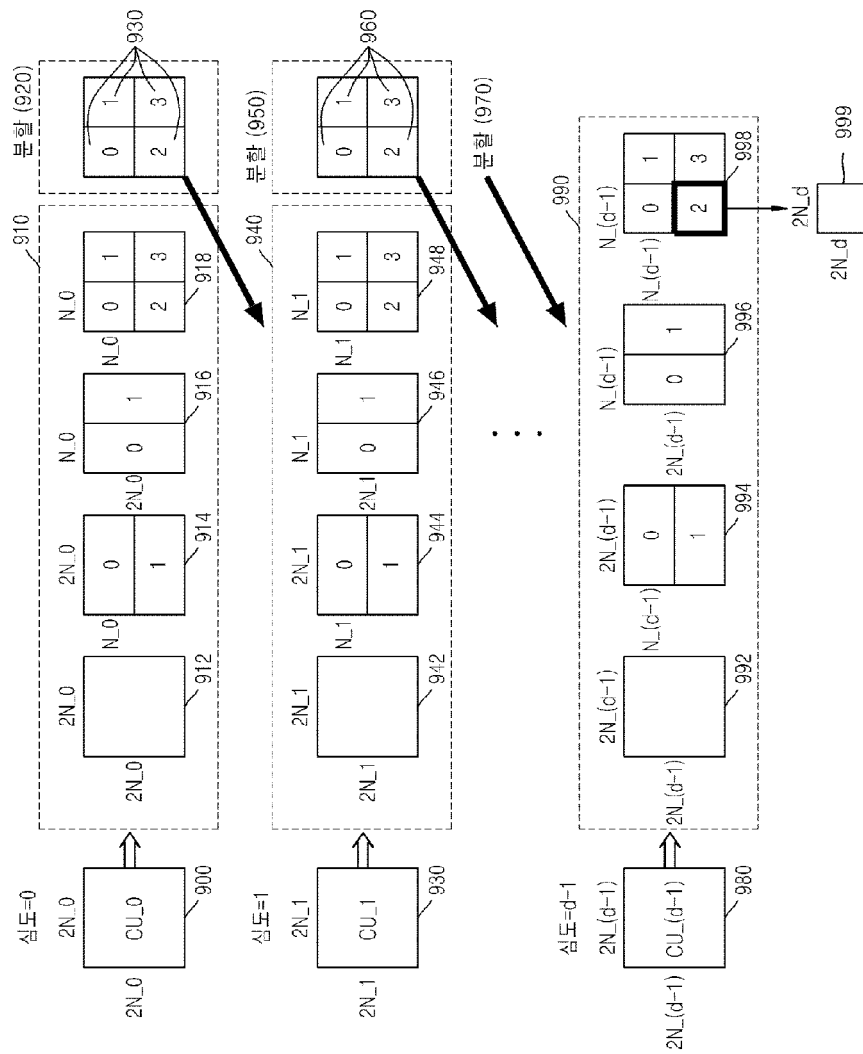
[Fig. 14]



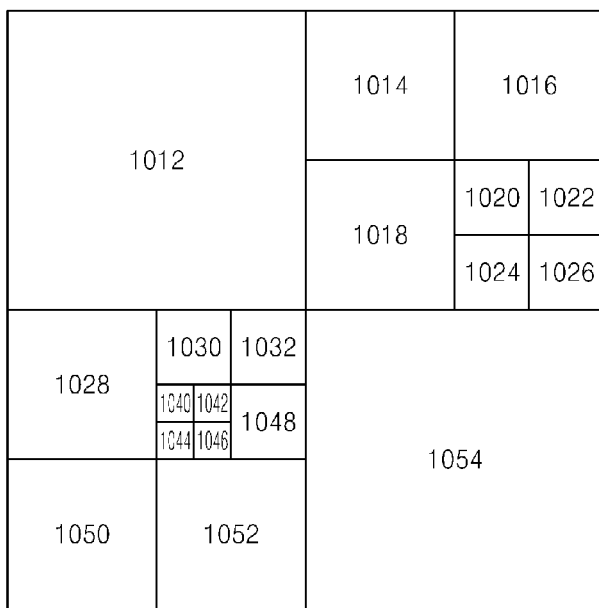
[Fig. 15]



[Fig. 16]

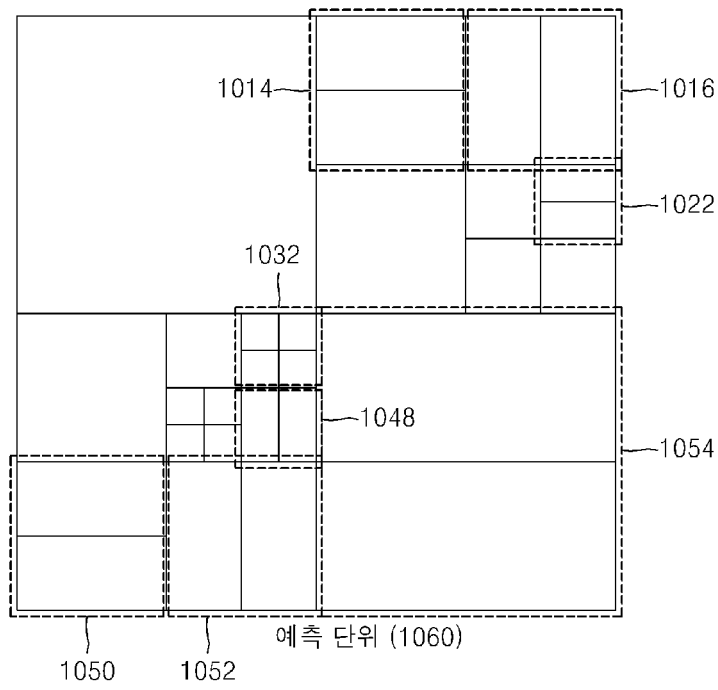


[Fig. 17]

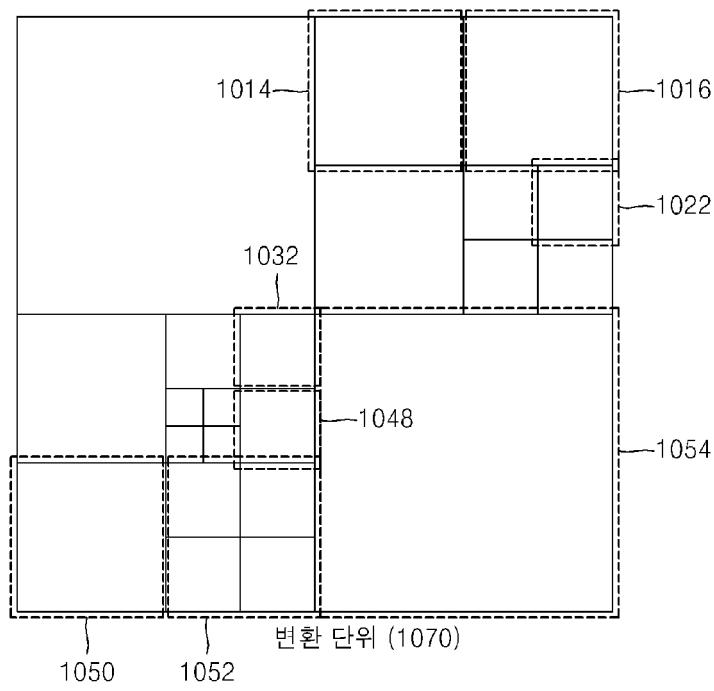


부호화 단위 (1010)

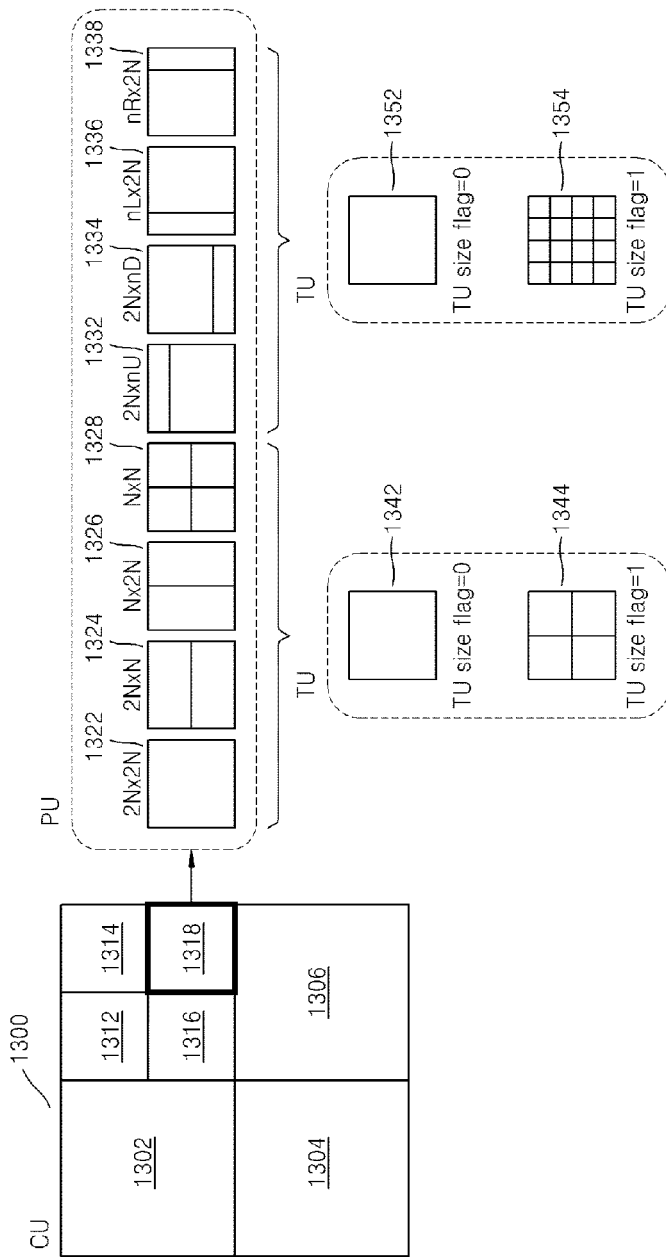
[Fig. 18]



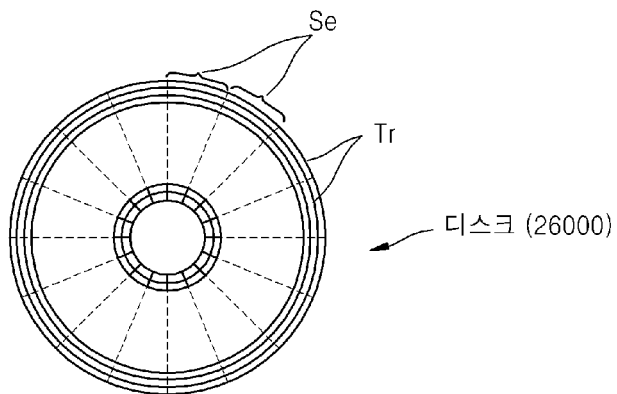
[Fig. 19]



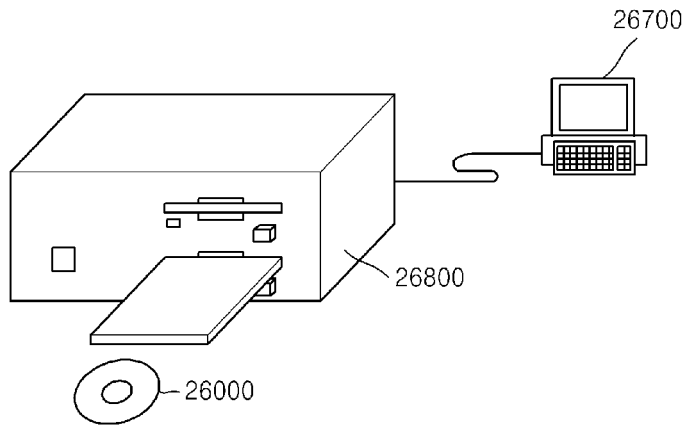
[Fig. 20]



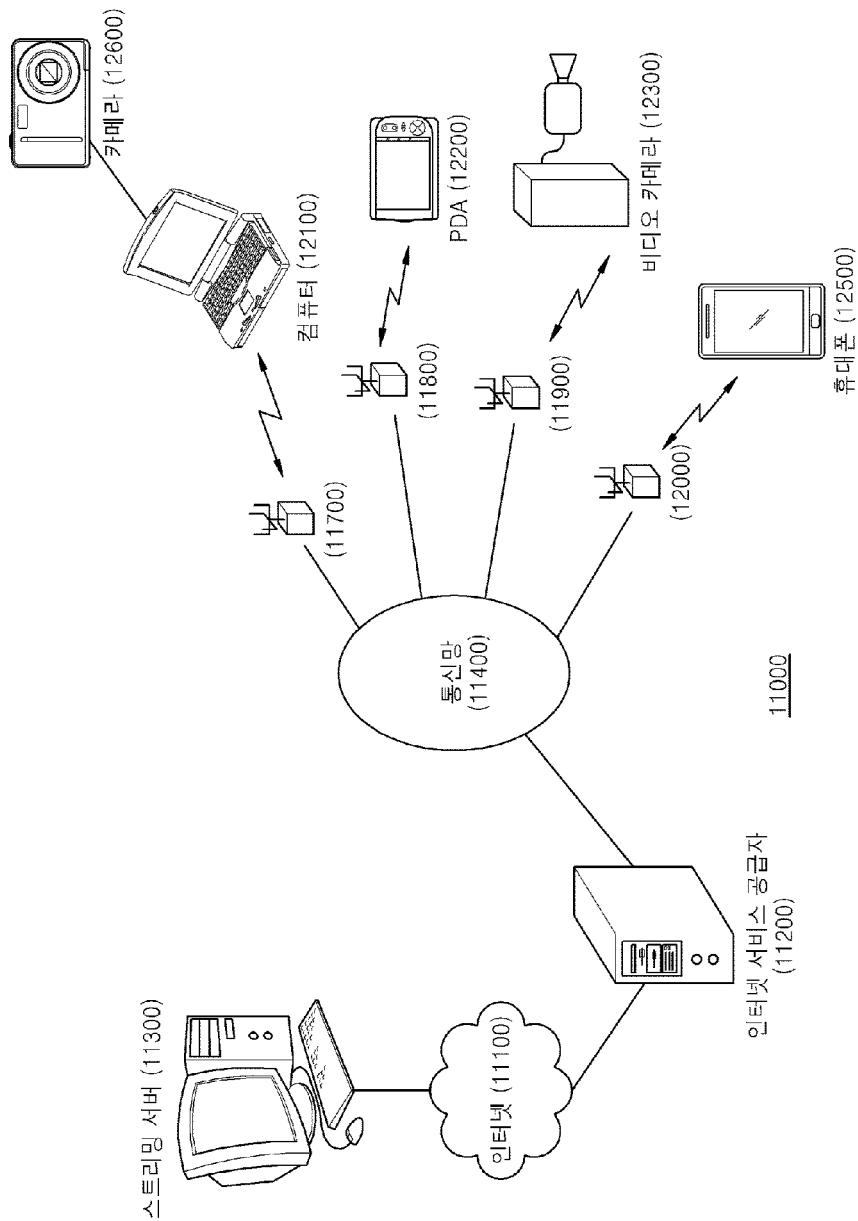
[Fig. 21]



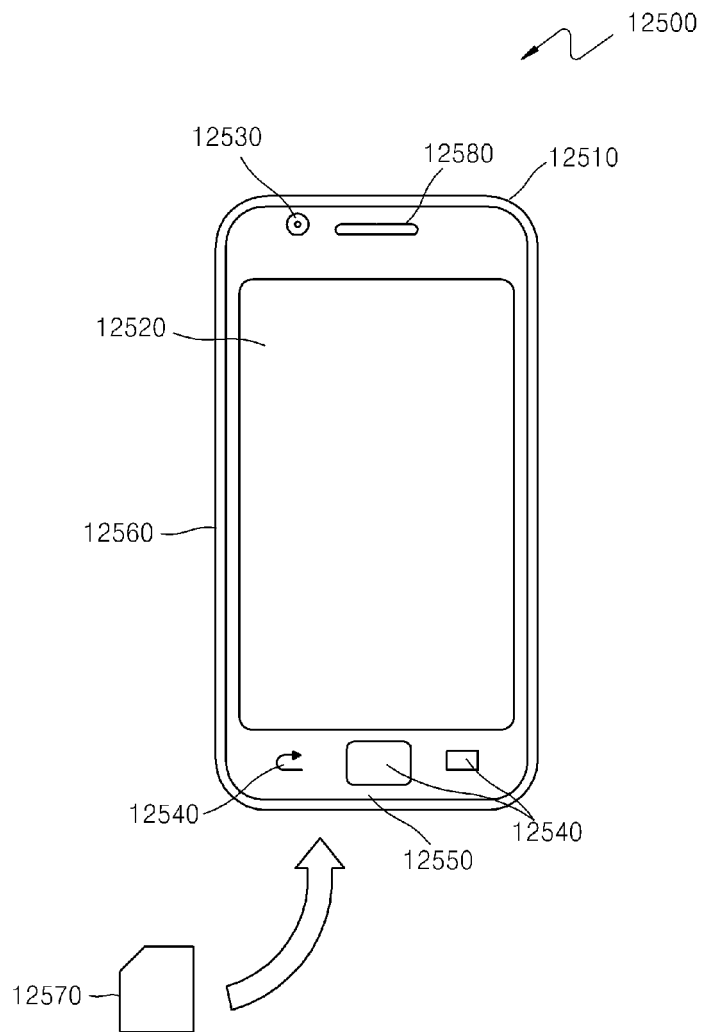
[Fig. 22]



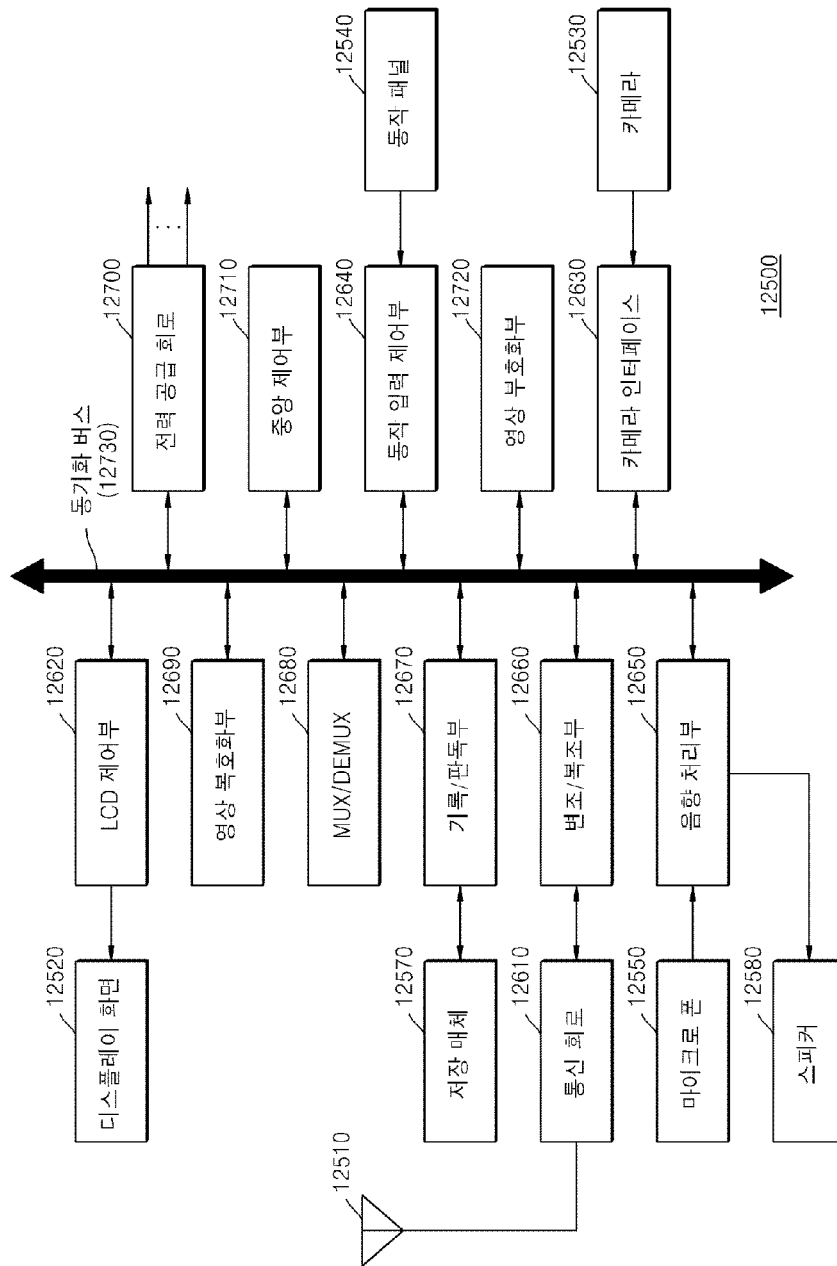
[Fig. 23]



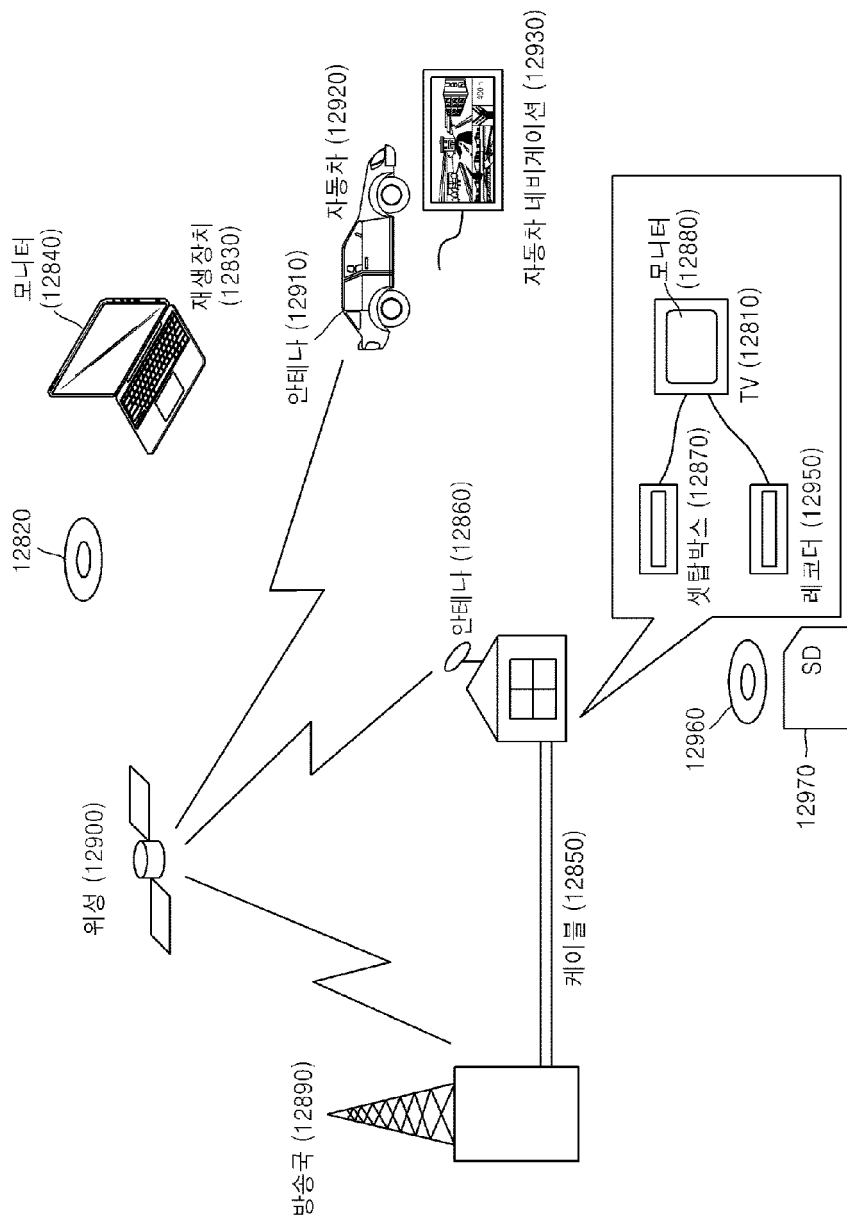
[Fig. 24]



[Fig. 25]



[Fig. 26]



[Fig. 27]

