

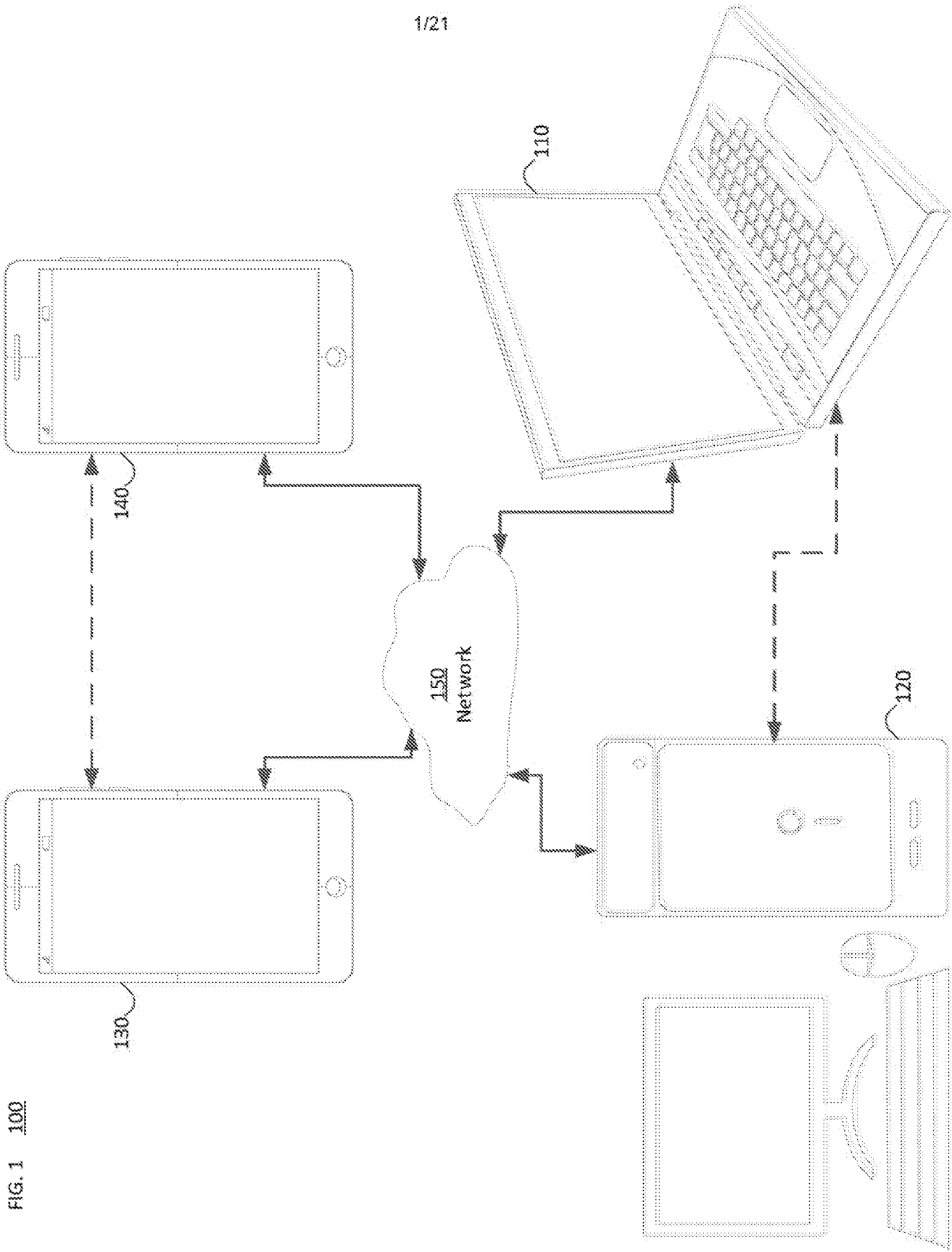


FIG. 1 100

FIG. 2
Streaming System 200

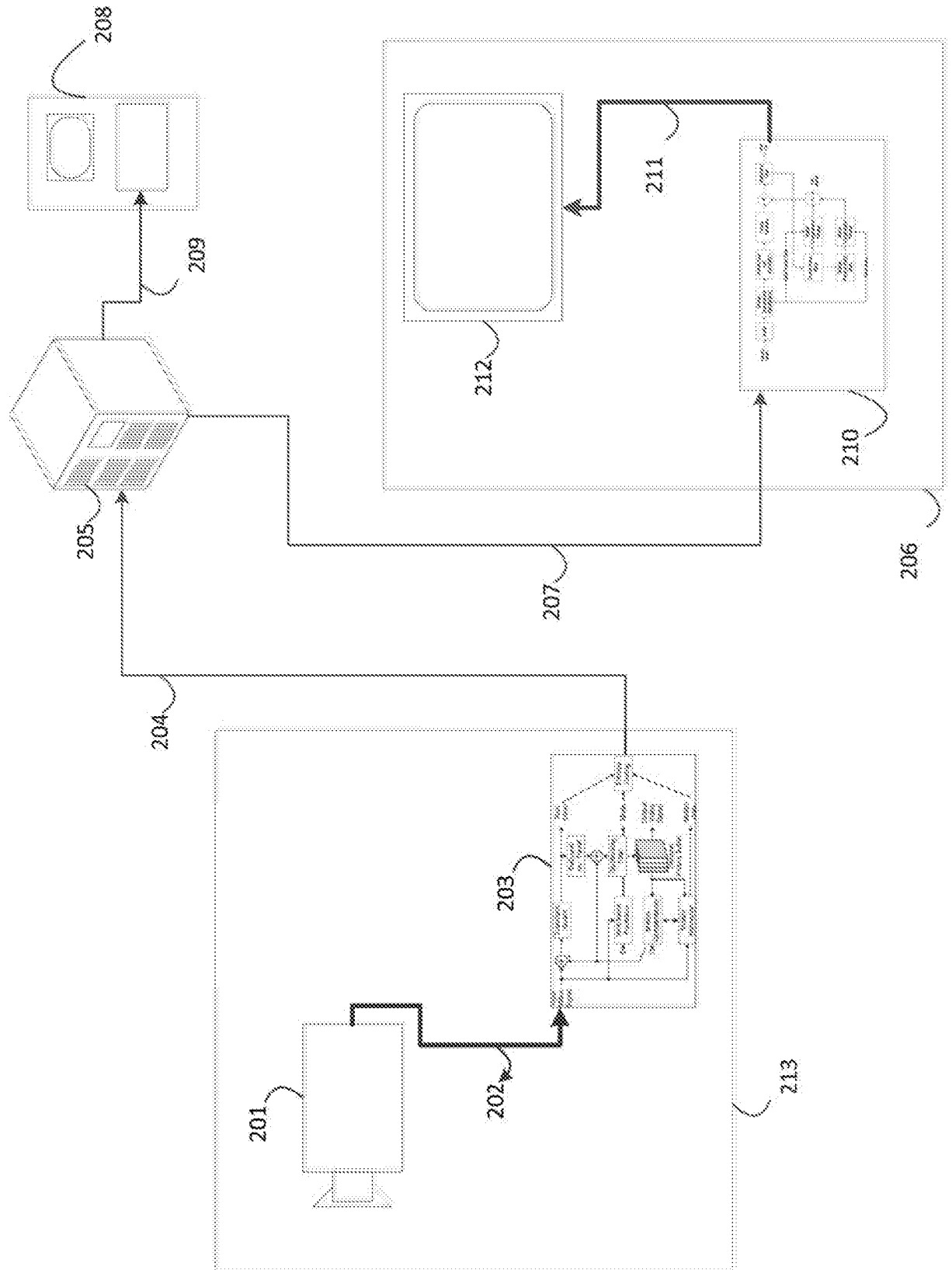


FIG. 3 Decoder 210

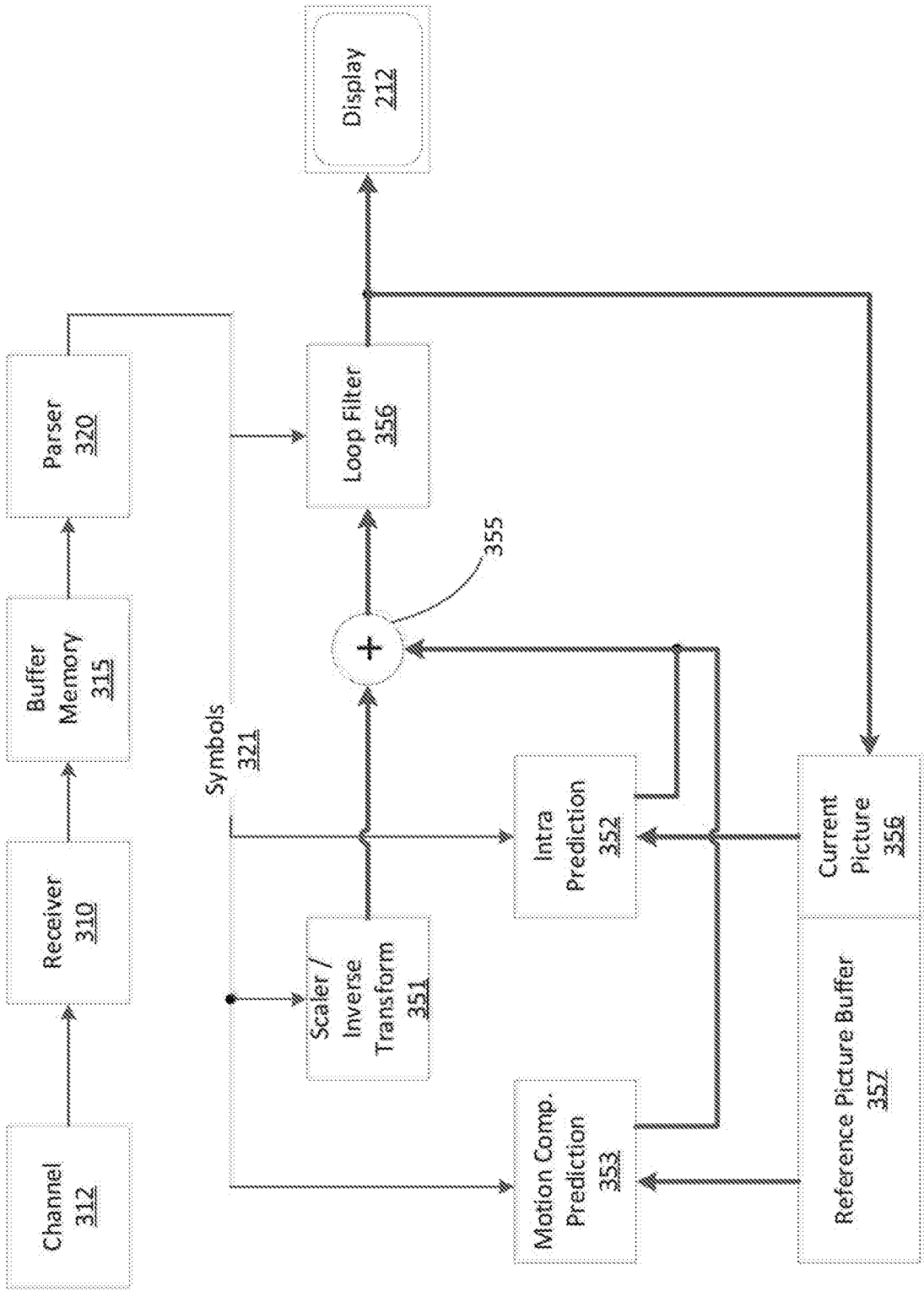


FIG. 4 Encoder 203

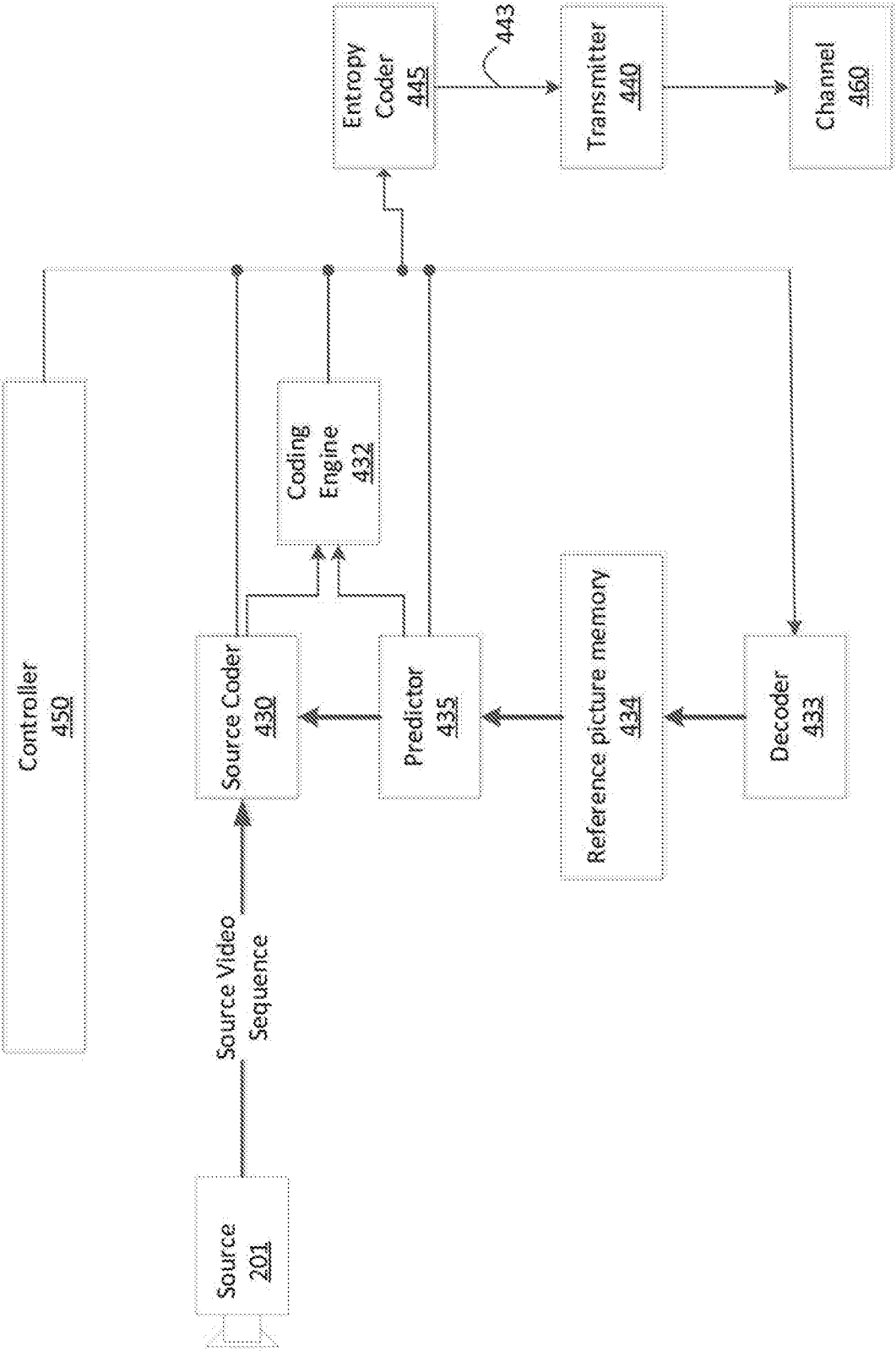
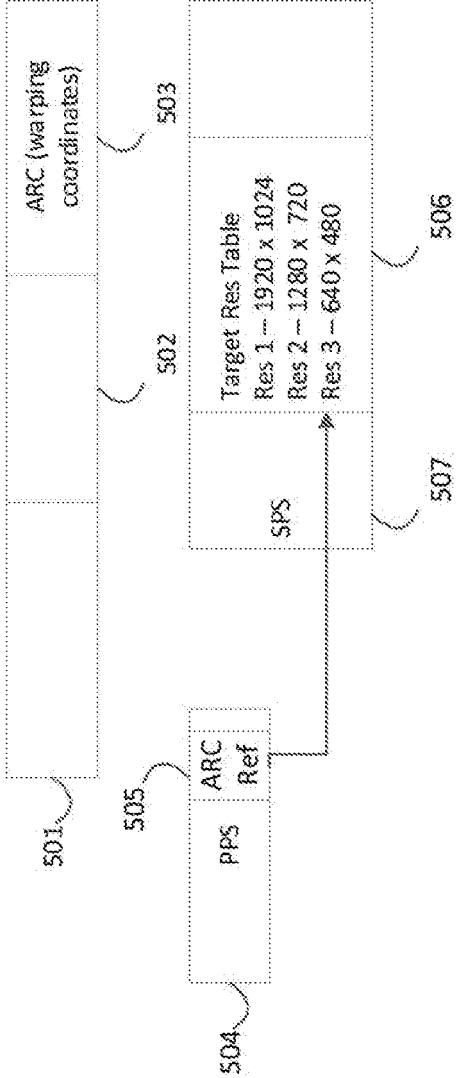


FIG. 5



Disclosed Subject
Matter

5/21

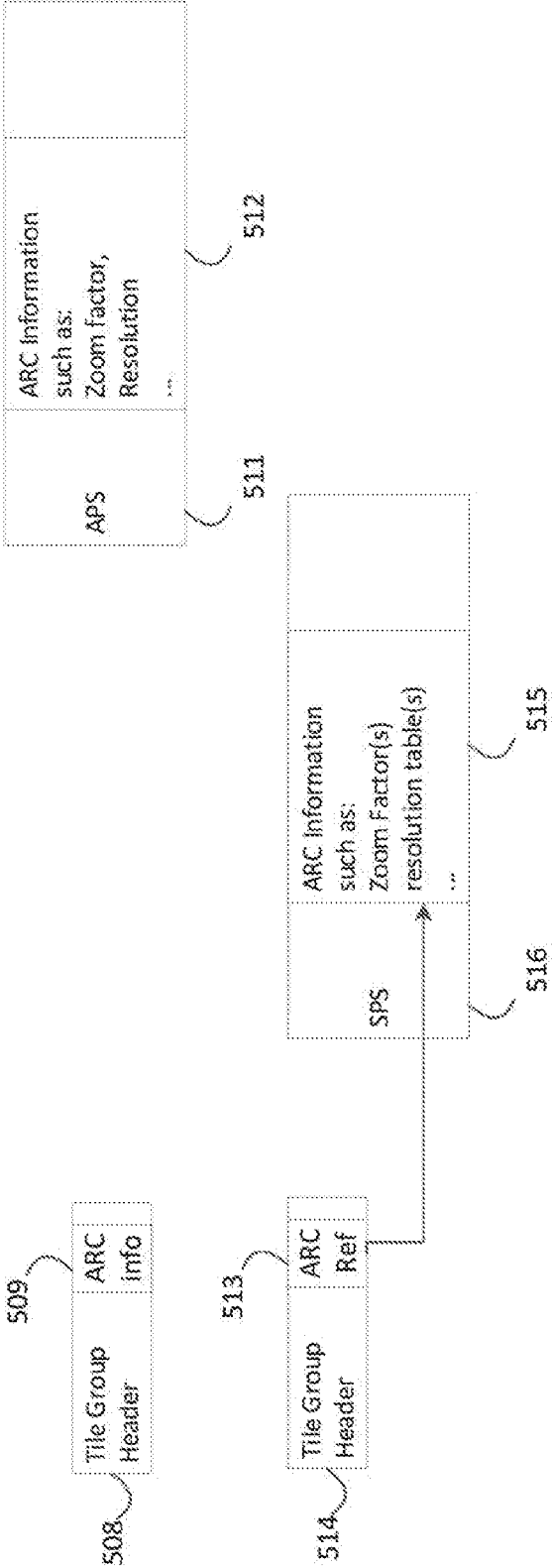


Fig. 6

601	tile_group_header() {		
	...		
603	if(adaptive_pic_resolution_change_flag) {		
602	dec_pic_size_idx		u(1)
	}		
	...		
	}		

610	seq_parameter_set_rbsp() {		Descriptor
	...		
611	adaptive_pic_resolution_change_flag		u(1)
612	if(adaptive_pic_resolution_change_flag) {		
613	output_pic_width_in_luma_samples		ue(v)
	output_pic_height_in_luma_samples		ue(v)
614	reference_pic_size_present_flag		u(1)
	if(reference_pic_size_present_flag)		
	{		
	reference_pic_width_in_luma_samples		ue(v)
615	reference_pic_height_in_luma_samples		ue(v)
	}		
616	num_dec_pic_size_in_luma_samples_minus1		ue(v)
	for(i = 0; i <= num_dec_pic_size_in_luma_samples_minus1; i++) {		
617	dec_pic_width_in_luma_samples[i]		ue(v)
	dec_pic_height_in_luma_samples[i]		ue(v)
	}		
	}		
	...		
	}		

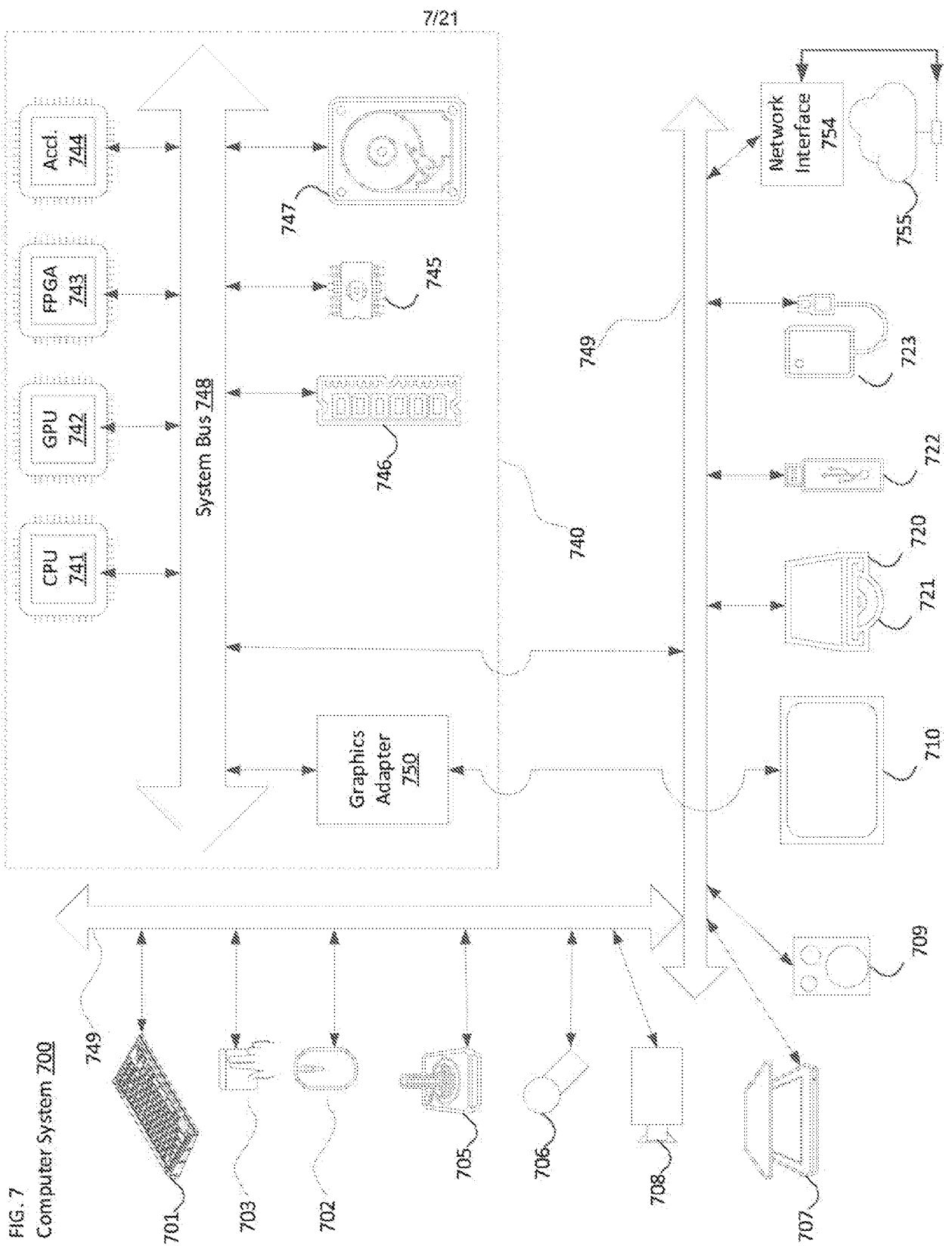
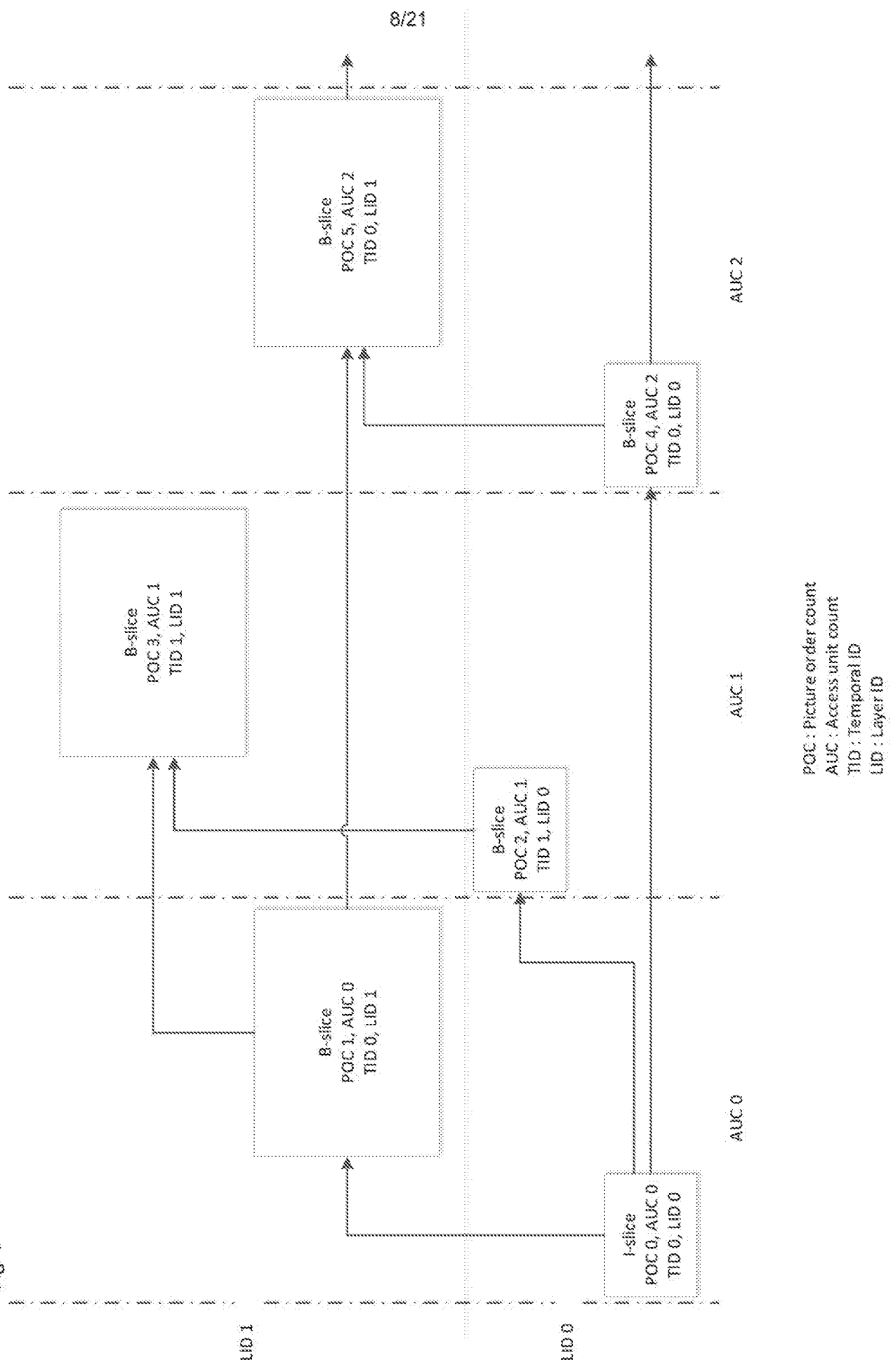


Fig. 8



9/21

Fig. 9

	Descriptor ¹
video_parameter_set_flag() { ²	
vps_video_parameter_set_id ³	u(4) ²
vps_max_layers_minus1 ⁴	u(8) ²
for(i = 0; i <= vps_max_layers_minus1; i++) { ⁵	²
vps_included_layer_id(i) ⁶	u(7) ²
vps_reserved_zero_bit ⁷	u(1) ²
} ⁸	²
vps_constraint_info_present_flag ⁹	u(1) ²
vps_constant_poc_cycle_per_aud ¹⁰	u(1) ²
if(vps_constant_poc_cycle_per_aud) ¹¹	²
vps_poc_cycle_aud ¹²	u(8) ²
... ¹³	²
} ¹⁴	²
¹⁵	
slice_header() { ¹⁶	
slice_pic_parameter_set_id ¹⁷	ue(v) ¹⁶
if(rect_slice_flag NumBricksInPic > 1) ¹⁸	¹⁶
slice_address ¹⁹	u(v) ¹⁶
if(!rect_slice_flag && !single_brick_per_slice_flag) ²⁰	¹⁶
num_bricks_in_slice_minus1 ²¹	ue(v) ¹⁶
slice_type ²²	ue(v) ¹⁶
if(NalUnitType == GRA_NUT) ²³	¹⁶
recovery_poc_cnt ²⁴	ue(v) ¹⁶
slice_pic_order_cnt_lsb ²⁵	u(v) ¹⁶
... ²⁶	¹⁶
if(vps_constant_poc_cycle_per_aud) ²⁷	¹⁶
slice_poc_cycle_aud ²⁸	u(8) ¹⁶
²⁹	¹⁶
³⁰	¹⁶
} ³¹	¹⁶

10/21

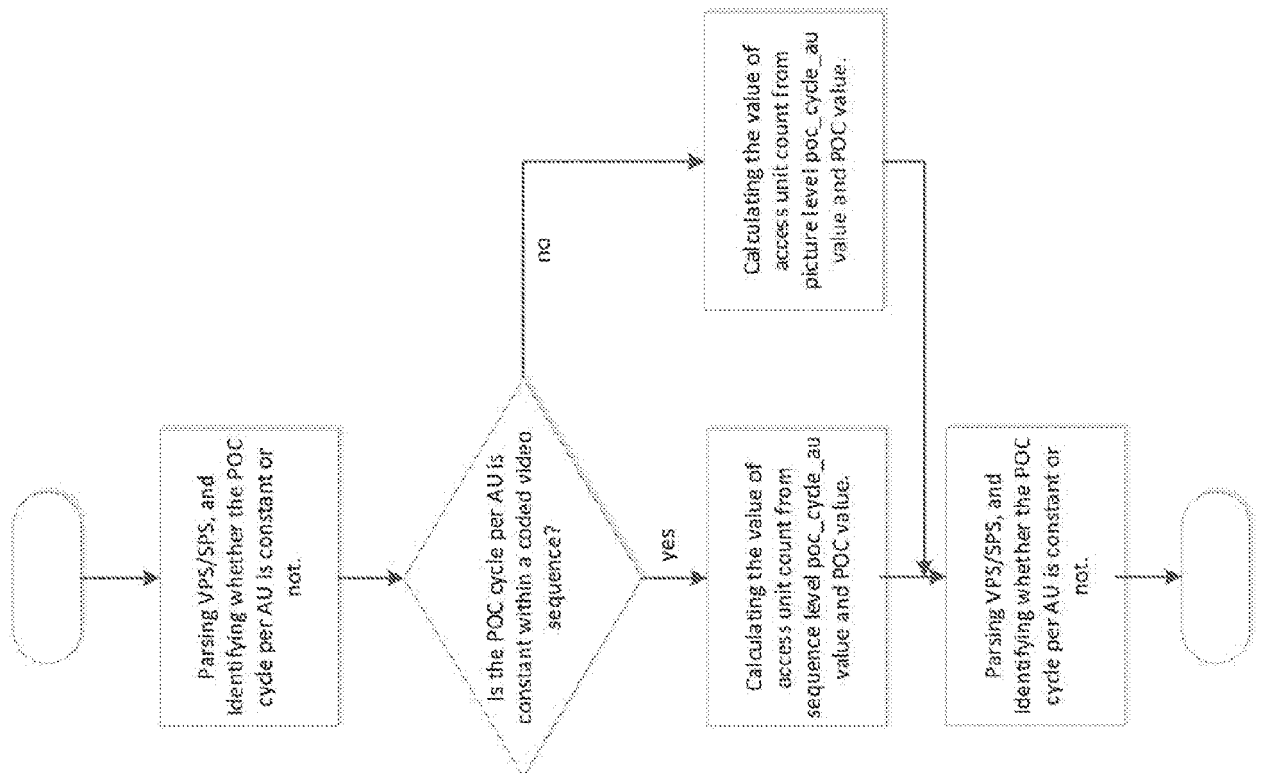
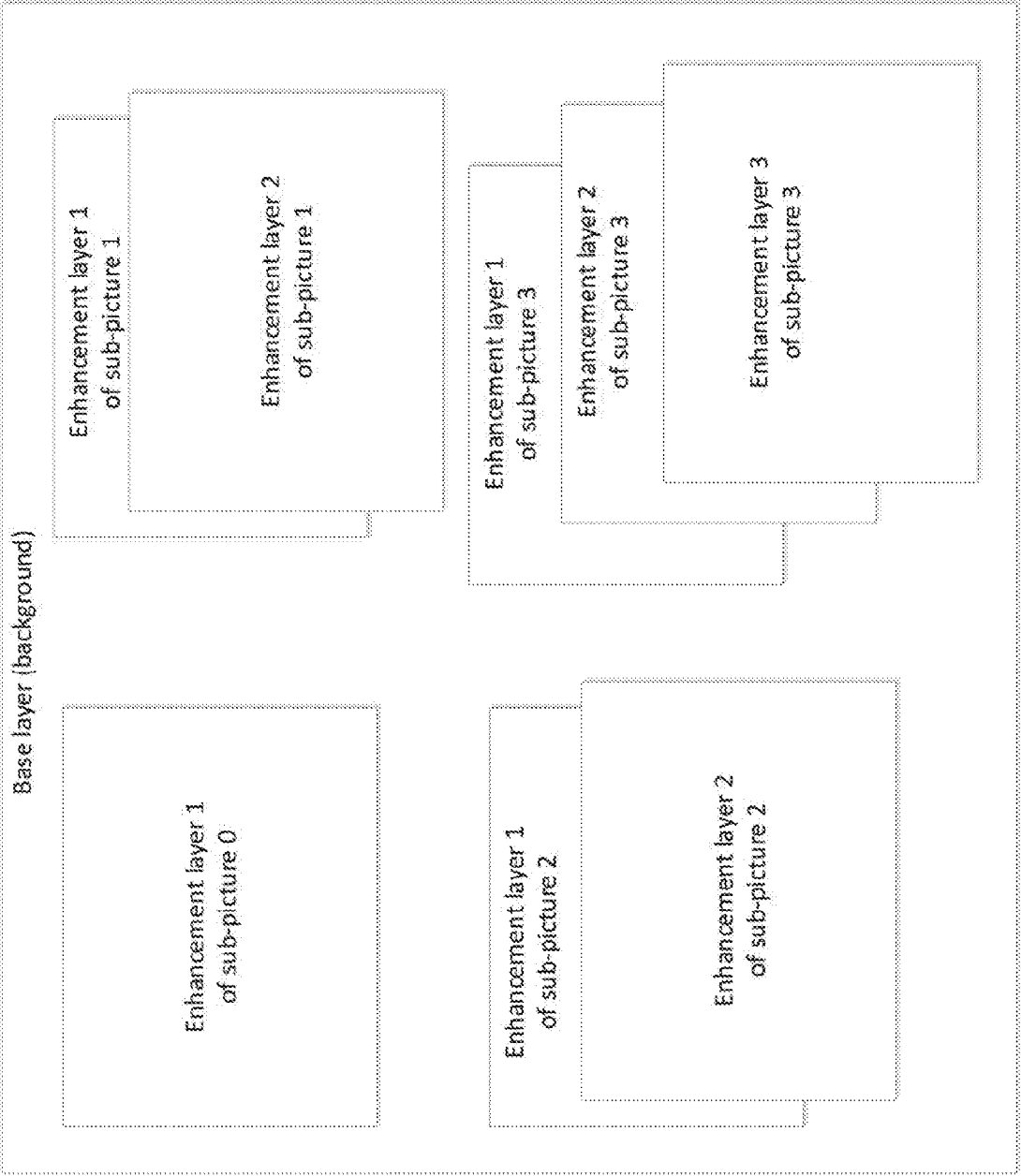


Fig. 10

FIG. 11



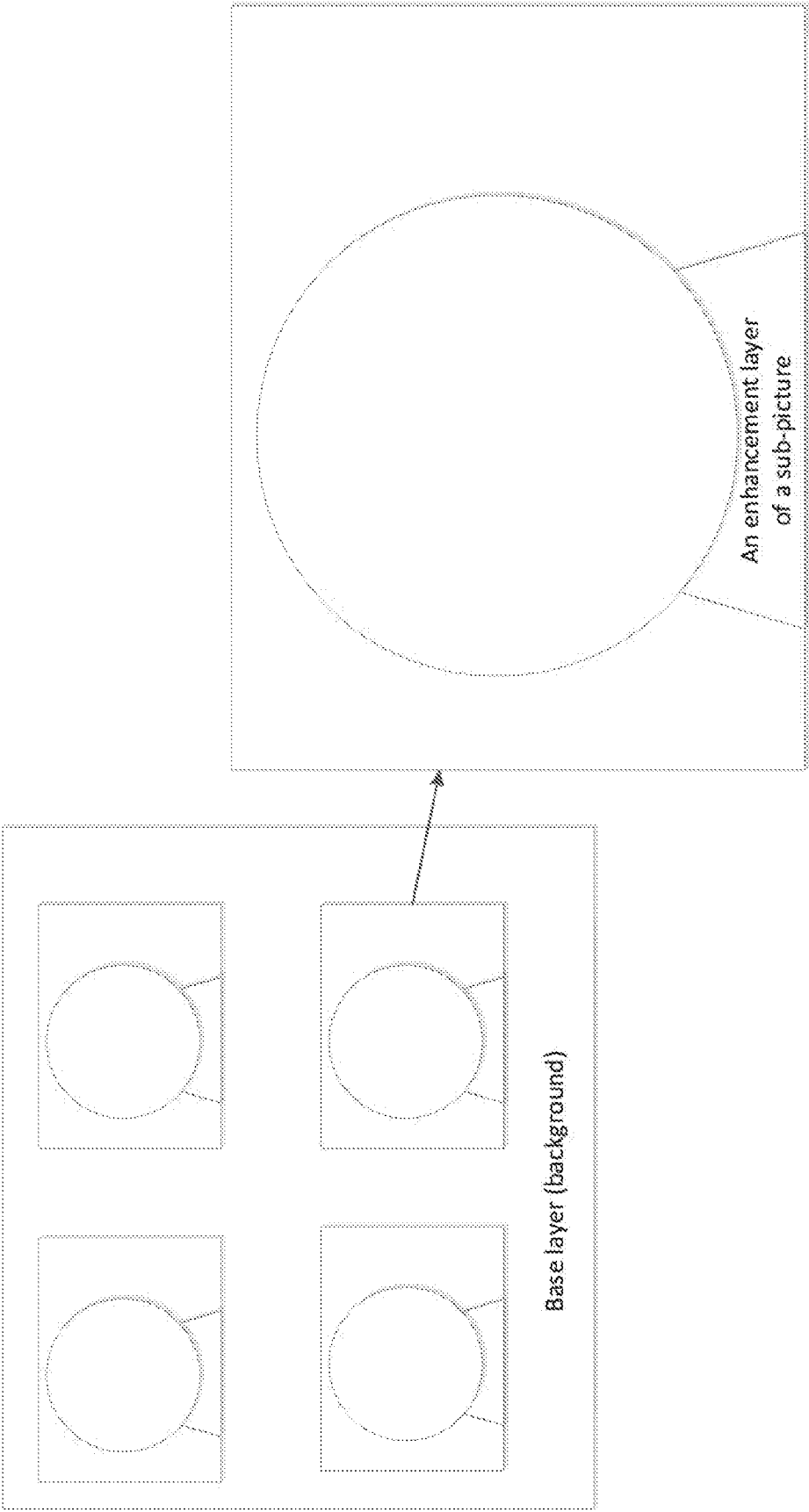
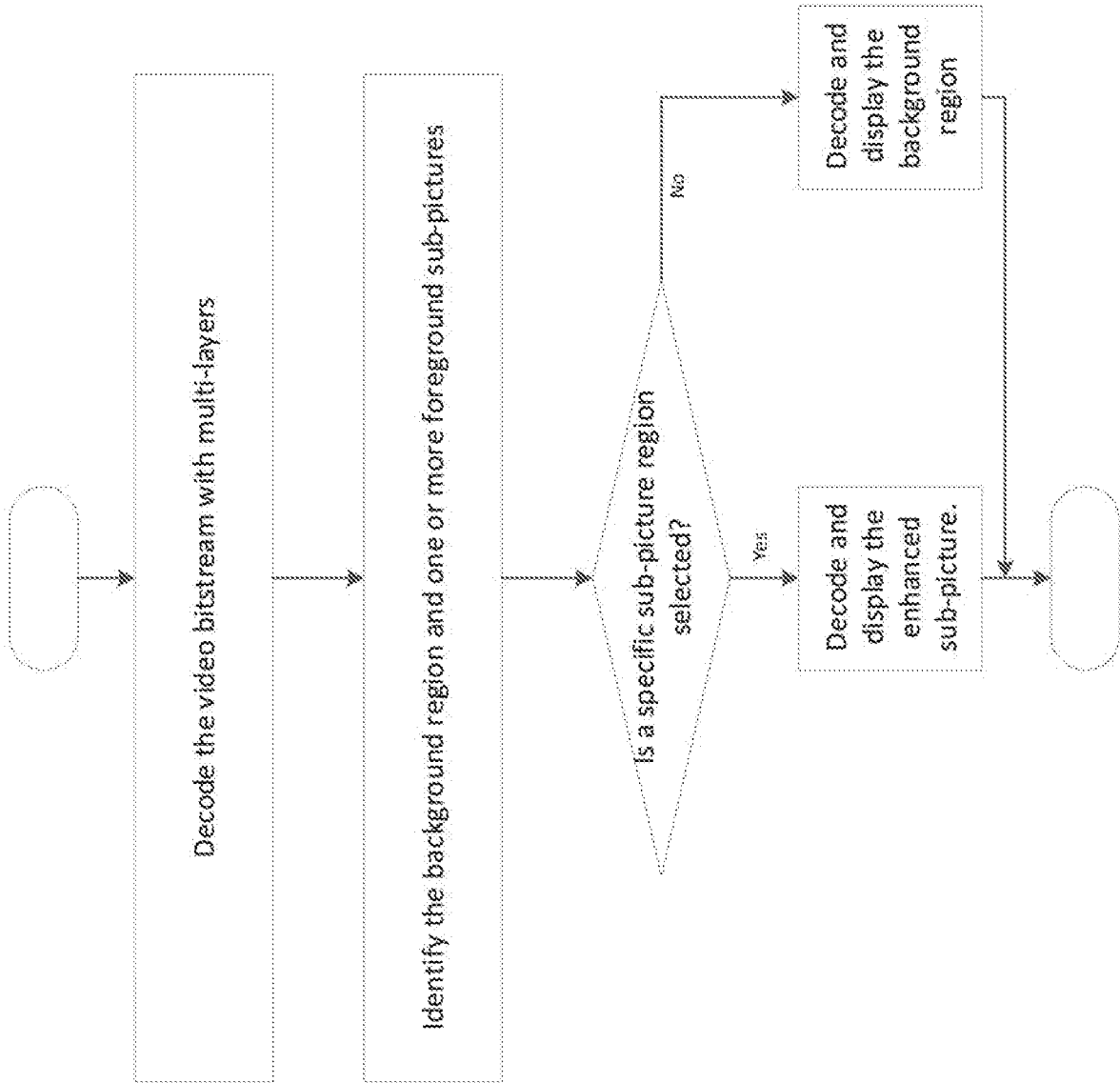


FIG. 12

Fig. 13



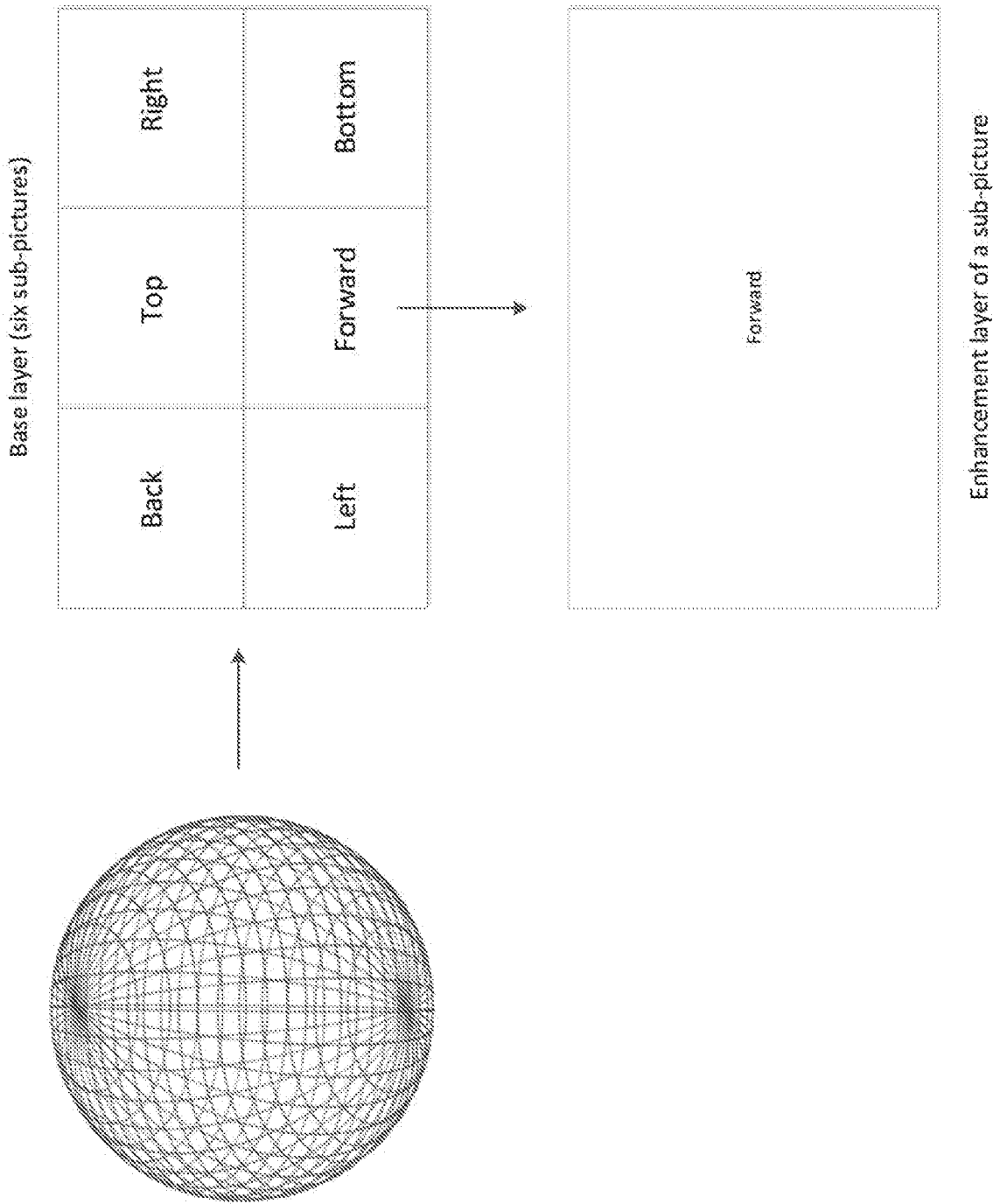


Fig. 14

Fig. 15

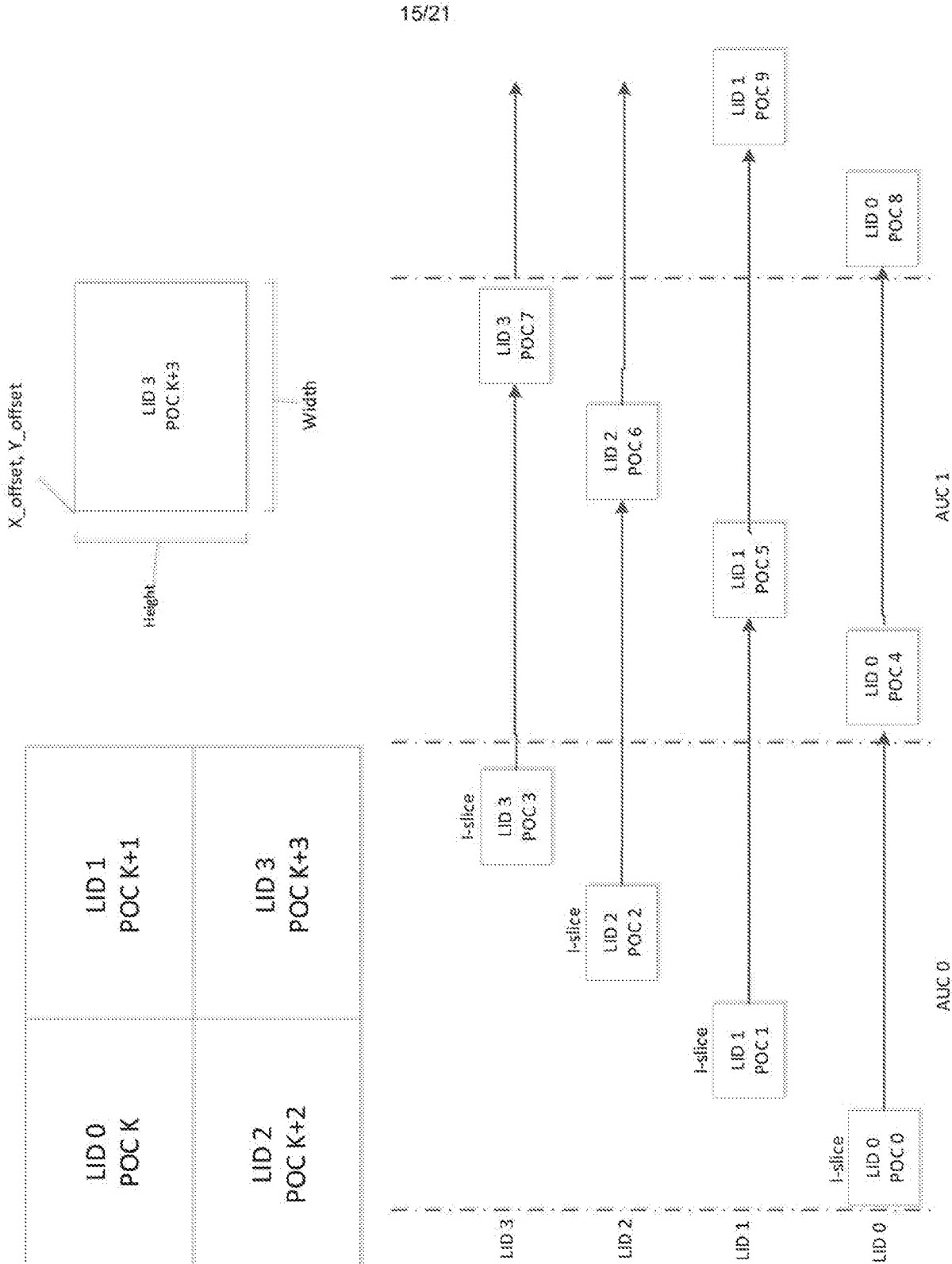
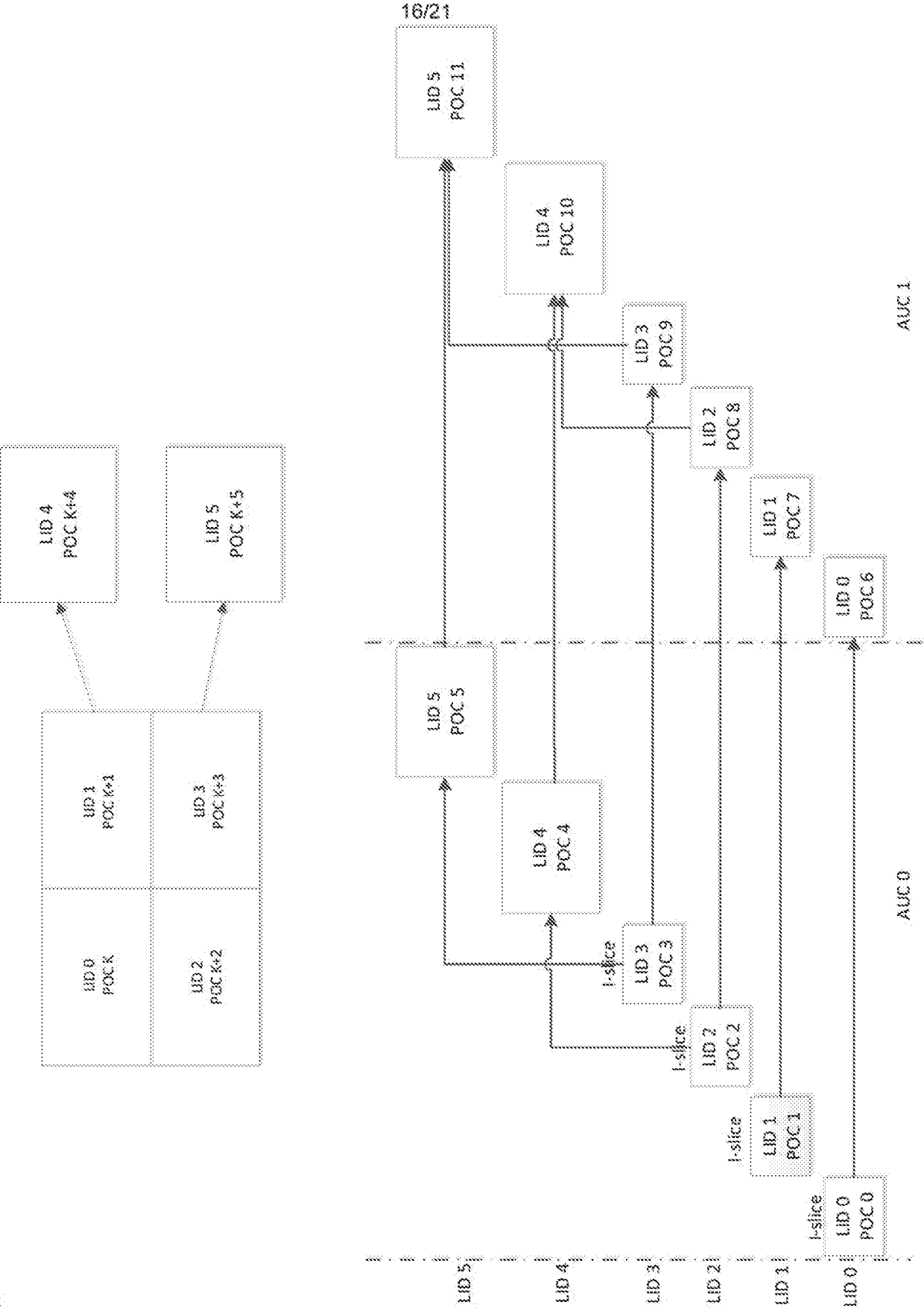


Fig. 16



17/21

Fig. 17

Descriptor	
video_parameters_set_rbsp() {	
xps_video_parameter_set_id	u(4)
vps_max_layers_minus1	u(8)
for(i = 0; i <= vps_max_layers_minus1; i++) {	
vps_included_layer_id[i]	u(7)
vps_reserved_zero_bit	u(1)
}	
...	
vps_sub_picture_dividing_flag	u(1)
if(vps_sub_picture_dividing_flag) {	
vps_full_pic_width_in_luma_samples	ue(v)
vps_full_pic_height_in_luma_samples	ue(v)
}	
...	
}	
•seq_parameter_set_rbsp() {	
sps_decoding_parameter_set_id	u(4)
sps_video_parameter_set_id	u(4)
sps_max_sub_layers_minus1	u(3)
...	
pic_width_in_luma_samples	ue(v)
pic_height_in_luma_samples	ue(v)
if(vps_sub_picture_dividing_flag) {	
pic_offset_x	ue(v)
pic_offset_y	ue(v)
}	
...	
}	

Fig. 18

Descriptor	
sub_region_partitioning_info(payloadSize) {	
num_sub_region {	ue(v)
num_layers {	ue(v)
for(j = 0; j <= num_layers; i++) {	
layer_id [i] {	ue(v)
for(i = 1; i < num_layers; i++) {	
for(j = 0; j < i; j++) {	
direct_dependency_flag[i][j] {	ue(1)
for(j = 0; j < num_sub_region; i++) {	
num_layers_for_region[i] {	ue(v)
for(j = 0; j < num_layers_for_region[i]; j++) {	
sub_region_layer_id[i][j] {	ue(v)
sub_region_offset_x[i] {	ue(v)
sub_region_offset_y[i] {	ue(v)
sub_region_width[i] {	ue(v)
sub_region_height [i] {	ue(v)
}	
... {	
}	
}	

Fig. 19

	Descriptor
video_parameter_set_rsp() {	
...	
vps_max_layers_minus1	u(6)
num_output_layer_sets	ue(v)
num_profile_tier_level	ue(v)
for(i = 0; i < num_profile_tier_level; i++)	
profile_tier_level(vps_max_sub_layers_minus1)	
for(i = 0; i < num_output_layer_sets; i++)	
for(j = 0; j < NumLayersInInitList(i); j++) {	
output_layer_flag[i][j]	u(1)
profile_tier_level_idx[i][j]	u(v)
}	
...	
}	

Fig. 20

	Descriptor
video_parameter_set_rbsp() {	
...	
vps_max_layers_minus1	u(6)
if(vps_max_layers_minus1 > 0) {	
num_output_layer_sets	ue(v)
num_profile_tier_level	ue(v)
}	
for(i = 0; i < num_profile_tier_level; i++)	
profile_tier_level(vps_max_sub_layers_minus1)	
for(i = 0; i < num_output_layer_sets; i++) {	
vps_output_layers_mode[i]	u(2)
vps_ptl_signal_flag[i]	u(1)
for(j = 0; j < NumLayersInHdList[i]; j++) {	
if(vps_output_layers_mode[i] == 2)	
output_layer_flag[i][j]	u(1)
if vps_ptl_signal_flag[i]	
profile_tier_level_idx[i][j]	u(v)
}	
}	
...	
}	

21/21

Fig. 21

video parameter set rsp() {	Descriptor
...	
vps_max_layers_minus1	u(6)
if(vps_max_layers_minus1 > 0) {	
num_output_layer_sets	ue(v)
num_profile_tier_level	ue(v)
}	
max_subpics_minus1	u(8)
for(i = 0; i < max_subpics_minus1; i++) {	
sub_pic_id[i]	u(8)
}	
for(i = 0; i < num_profile_tier_level; i++)	
profile_tier_level[vps_max_sub_layers_minus1]	
for(i = 0; i < num_output_layer_sets; i++) {	
vps_output_layers_mode[i]	u(2)
vps_ptl_signal_flag[i]	u(1)
for(j = 0; j < NumLayersInIdList(i); j++) {	
num_output_subpic_layer[i][j]	ue(v)
for(k = 0; k < num_output_subpic_layer[i][j]; k++)	
sub_pic_id_layer[i][j][k]	u(8)
if(vps_output_layers_mode[i] == 2)	
output_layer_flag[i][j]	u(1)
if vps_ptl_signal_flag[i]	
profile_tier_level_idn[i][j]	u(v)
}	
}	
...	
}	