

【H.264/AVC视频编解码技术详解】 十、图像参数集Picture Paramater Set(PPS)解析

 blog.csdn.net/shagoneal/article/details/52877689

《H.264/AVC视频编解码技术详解》视频教程已经在“CSDN学院”上线，视频中详述了H.264的背景、标准协议和实现，并通过一个实战工程的形式对H.264的标准进行解析和实现，欢迎观看！

“纸上得来终觉浅，绝知此事要躬行”，只有自己按照标准文档以代码的形式操作一遍，才能对视频压缩编码标准的思想和方法有足够深刻的理解和体会！

链接地址：[H.264/AVC视频编解码技术详解](#)

GitHub代码地址：[点击这里](#)

除了序列参数集SPS之外，H.264中另一重要的参数集合为图像参数集Picture Paramater Set(PPS)。通常情况下，PPS类似于SPS，在H.264的裸码流中单独保存在一个NAL Unit中，只是PPS NAL Unit的nal_unit_type值为8；而在封装格式中，PPS通常与SPS一起，保存在视频文件的文件头中。

1. 图像参数集PPS的定义

在H.264的协议文档中，PPS的结构定义在7.3.2.2节中，具体的结构如下表所示：

pic_parameter_set_rbsp() {	C	Descriptor
pic_parameter_set_id	1	ue(v)
seq_parameter_set_id	1	ue(v)
entropy_coding_mode_flag	1	u(1)
bottom_field_pic_order_in_frame_present_flag	1	u(1)
num_slice_groups_minus1	1	ue(v)
if(num_slice_groups_minus1 > 0) {		
slice_group_map_type	1	ue(v)
if(slice_group_map_type == 0)		
for(iGroup = 0; iGroup <= num_slice_groups_minus1; iGroup++)		
run_length_minus1[iGroup]	1	ue(v)
else if(slice_group_map_type == 2)		
for(iGroup = 0; iGroup < num_slice_groups_minus1; iGroup++) {		
top_left[iGroup]	1	ue(v)
bottom_right[iGroup]	1	ue(v)
}		
else if(slice_group_map_type == 3 slice_group_map_type == 4 slice_group_map_type == 5) {		
slice_group_change_direction_flag	1	u(1)
slice_group_change_rate_minus1	1	ue(v)
} else if(slice_group_map_type == 6) {		
pic_size_in_map_units_minus1	1	ue(v)
for(i = 0; i <= pic_size_in_map_units_minus1; i++)		
slice_group_id[i]	1	u(v)
}		
}		

<https://blog.csdn.net/shaqoneal>

2. PPS中的语法元素

(1). pic_parameter_set_id

表示当前PPS的id。某个PPS在码流中会被相应的slice引用，slice引用PPS的方式就是在Slice header中保存PPS的id值。该值的取值范围为[0,255]。

(2). seq_parameter_set_id

表示当前PPS所引用的激活的SPS的id。通过这种方式，PPS中也可以取到对应SPS中的参数。该值的取值范围为[0,31]。

(3). entropy_coding_mode_flag

熵编码模式标识，该标识位表示码流中熵编码/解码选择的算法。对于部分语法元素，在不同的编码配置下，选择的熵编码方式不同。例如在一个宏块语法元素中，宏块类型mb_type的语法元素描述符为“ue(v) | ae(v)”，在baseline profile等设置下采用指数哥伦布编码，在main

profile等设置下采用CABAC编码。

标识位entropy_coding_mode_flag的作用就是控制这种算法选择。当该值为0时，选择左边的算法，通常为指数哥伦布编码或者CAVLC；当该值为1时，选择右边的算法，通常为CABAC。

(4). bottom_field_pic_order_in_frame_present_flag

标识位，用于表示另外条带头中的两个语法元素delta_pic_order_cnt_bottom和delta_pic_order_cn是否存在的标识。这两个语法元素表示了某一帧的底场的POC的计算方法。

(5). num_slice_groups_minus1

表示某一帧中slice group的个数。当该值为0时，一帧中所有的slice都属于一个slice group。slice group是一帧中宏块的组合方式，定义在协议文档的3.141部分。

(6). num_ref_idx_l0_default_active_minus1、 num_ref_idx_l1_default_active_minus1

表示当Slice Header中的num_ref_idx_active_override_flag标识位为0时，P/SP/B slice的语法元素num_ref_idx_l0_active_minus1和num_ref_idx_l1_active_minus1的默认值。

(7). weighted_pred_flag

标识位，表示在P/SP slice中是否开启加权预测。

(8). weighted_bipred_idc

表示在B Slice中加权预测的方法，取值范围为[0,2]。0表示默认加权预测，1表示显式加权预测，2表示隐式加权预测。

(9). pic_init_qp_minus26和pic_init_qs_minus26

表示初始的量化参数。实际的量化参数由该参数、slice header中的slice_qp_delta/slice_qs_delta计算得到。

(10). chroma_qp_index_offset

用于计算色度分量的量化参数，取值范围为[-12,12]。

(11). deblocking_filter_control_present_flag

标识位，用于表示Slice header中是否存在用于去块滤波器控制的信息。当该标志位为1时，slice header中包含去块滤波相应的信息；当该标识位为0时，slice header中没有相应的信息。

(12). constrained_intra_pred_flag

若该标识为1，表示I宏块在进行帧内预测时只能使用来自I和SI类型宏块的信息；若该标识位0，表示I宏块可以使用来自Inter类型宏块的信息。

(13). redundant_pic_cnt_present_flag

标识位，用于表示Slice header中是否存在redundant_pic_cnt语法元素。当该标志位为1时，slice header中包含redundant_pic_cnt；当该标识位为0时，slice header中没有相应的信息。