

# ffmpeg推流 av\_dict\_set 参数设置解析（降低延时、处理花屏、改善画面质量）（实时更新）

原创

ZackZheng999

已于 2022-03-21

14:53:58 修改

2710

★ 收藏

版权

21

分类专栏： ffmpeg

文章标签： ffmpeg



ffmpeg 专栏收录该内容

2 订阅

5 篇文章

订阅专栏

## 一、关于影响时延的参数设置

**1.preset**：指定编码的配置。x264编码算法有很多可供配置参数，不同的参数值会导致编码的速度大相径庭，甚至可能影响质量。为了免去用户了解算法，然后手工配置参数的麻烦。x264提供了一些预设值，而这些预设值可以通过preset指定。这些预设值有包括：ultrafast, superfast, veryfast, faster, fast, medium, slow, slower, veryslow和placebo。ultrafast编码速度最快，但压缩率低，生成的文件更大，placebo则正好相反。x264所取的默认值为medium。需要说明的是，preset主要是影响编码的速度，并不会很大的影响编码出来的结果的质量。压缩高清电影时，我一般用slow或者slower，当你的机器性能很好时也可以使用veryslow，不过一般并不会带来很大的好处。

例：`av_dict_set(&param, "preset", "ultrafast", 0);`

**2.muxdelay || max\_delay**：设置延迟约束，muxdelay以秒为单位设置延迟，而max\_delay以微秒为单位设置延迟。最终结果是相同的。

例：`av_dict_set(&param, "muxdelay", "1", 0);`

**3.zerolatency**：转码延迟，以牺牲视频质量减少时延

例：`av_dict_set(&param, "tune", "zerolatency", 0)`

## 二、关于影响视频质量的参数设置

**1.crf**：这是最重要的一个选项，用于指定输出视频的质量，取值范围是0-51，默认值为23，数字越小输出视频的质量越高。这个选项会直接影响到输出视频的码率。一般来说，压制480p我会用20左右，压制720p我会用16-18，1080p我会用12。最好的办法是大家可以多尝试几个值，每个都压几分钟，看看最后的输出质量和文件大小，自己再按需选择。

该方法与设置编码器的bit\_rate的效果是一样的。

例：`av_dict_set(&param, "crf", "18", 0);`

### 2.profile

H.264有四种画质级别，分别是baseline, extended, main, high：

（1）Baseline Profile：基本画质。支持I/P 帧，只支持无交错（Progressive）和CAVLC；

（2）Extended profile：进阶画质。支持I/P/B/SP/SI 帧，只支持无交错（Progressive）和CAVLC；（用的少）

（3）Main profile：主流画质。提供I/P/B 帧，支持无交错（Progressive）和交错（Interlaced），也支持CAVLC 和CABAC 的支持；

（4）High profile：高级画质。在main Profile 的基础上增加了8x8内部预测、自定义量化、无损视频编码和更多的YUV 格式；

H.264 Baseline profile、Extended profile和Main profile都是针对8位样本数据、4:2:0格式(YUV)的视频序列。在相同配置情况下，High profile（HP）可以比Main profile（MP）降低10%的码率。根据应用领域的不同，Baseline profile多应用于实时通信领域，Main profile多应用于存储领域。



ZackZheng999

关注

0



	Baseline	Extended	Main	High	High 10	High 4:2:2	High 4:4:4 Predictive
I and P Slices	Yes	Yes	Yes	Yes	Yes	Yes	Yes
B Slices	No	Yes	Yes	Yes	Yes	Yes	Yes
SI and SP Slices	No	Yes	No	No	No	No	No
Multiple Reference Frames	Yes	Yes	Yes	Yes	Yes	Yes	Yes
In-Loop Deblocking Filter	Yes	Yes	Yes	Yes	Yes	Yes	Yes
CAVLC Entropy Coding	Yes	Yes	Yes	Yes	Yes	Yes	Yes
CABAC Entropy Coding	No	No	Yes	Yes	Yes	Yes	Yes
Flexible Macroblock Ordering (FMO)	Yes	Yes	No	No	No	No	No
Arbitrary Slice Ordering (ASO)	Yes	Yes	No	No	No	No	No
Redundant Slices (RS)	Yes	Yes	No	No	No	No	No
Data Partitioning	No	Yes	No	No	No	No	No
Interlaced Coding (PicAFF, MBAFF)	No	Yes	Yes	Yes	Yes	Yes	Yes
4:2:0 Chroma Format	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Monochrome Video Format (4:0:0)	No	No	No	Yes	Yes	Yes	Yes
4:2:2 Chroma Format	No	No	No	No	No	Yes	Yes
4:4:4 Chroma Format	No	No	No	No	No	No	Yes
8 Bit Sample Depth	Yes	Yes	Yes	Yes	Yes	Yes	Yes
9 and 10 Bit Sample Depth	No	No	No	No	Yes	Yes	Yes
11 to 14 Bit Sample Depth	No	No	No	No	No	No	Yes
8x8 vs. 4x4 Transform Adaptivity	No	No	No	Yes	Yes	Yes	Yes
Quantization Scaling Matrices	No	No	No	Yes	Yes	Yes	Yes
Separate Cb and Cr QP control	No	No	No	Yes	Yes	Yes	Yes
Separate Color Plane Coding	No	No	No	No	No	No	Yes
Predictive Lossless Coding	No	No	No	No	No	No	Yes
	Baseline	Extended	Main	High	High 10	High 4:2:2	High 4:4:4 Predictive

### 3.level:

level越高视频质量也就越高

例: `av_dict_set(&param, "level", "4", 0);`



Level number	Max macroblocks per second	Max frame size (macroblocks)	Max video bit rate (VCL) for Baseline, Extended and Main Profiles	Max video bit rate (VCL) for High Profile	Max video bit rate (VCL) for High 10 Profile	Max video bit rate (VCL) for High 4:2:2 and High 4:4:4 Predictive Profiles	Examples for high resolution @ frame rate (max stored frames) in Level
1	1485	99	64 kbit/s	80 kbit/s	192 kbit/s	256 kbit/s	128x96@30.9 (8) 176x144@15.0 (4)
1b	1485	99	128 kbit/s	160 kbit/s	384 kbit/s	512 kbit/s	128x96@30.9 (8) 176x144@15.0 (4)
1.1	3000	396	192 kbit/s	240 kbit/s	576 kbit/s	768 kbit/s	176x144@30.3 (9) 320x240@10.0 (3) 352x288@7.5 (2)
1.2	6000	396	384 kbit/s	480 kbit/s	1152 kbit/s	1536 kbit/s	320x240@20.0 (7) 352x288@15.2 (6)
1.3	11880	396	768 kbit/s	960 kbit/s	2304 kbit/s	3072 kbit/s	320x240@36.0 (7) 352x288@30.0 (6)
2	11880	396	2 Mbit/s	2.5 Mbit/s	6 Mbit/s	8 Mbit/s	320x240@36.0 (7) 352x288@30.0 (6)
2.1	19800	792	4 Mbit/s	5 Mbit/s	12 Mbit/s	16 Mbit/s	352x480@30.0 (7) 352x576@25.0 (6)
2.2	20250	1620	4 Mbit/s	5 Mbit/s	12 Mbit/s	16 Mbit/s	352x480@30.7 (10) 352x576@25.6 (7) 720x480@15.0 (6) 720x576@12.5 (5)
3	40500	1620	10 Mbit/s	12.5 Mbit/s	30 Mbit/s	40 Mbit/s	352x480@61.4 (12) 352x576@51.1 (10) 720x480@30.0 (6) 720x576@25.0 (5)
3.1	108000	3600	14 Mbit/s	17.5 Mbit/s	42 Mbit/s	56 Mbit/s	720x480@80.0 (13) 720x576@66.7 (11) 1280x720@30.0 (5)
3.2	216000	5120	20 Mbit/s	25 Mbit/s	60 Mbit/s	80 Mbit/s	1280x720@60.0 (5) 1280x1024@42.2 (4)
4	245760	8192	20 Mbit/s	25 Mbit/s	60 Mbit/s	80 Mbit/s	1280x720@68.3 (9) 1920x1088@30.1 (4) 2048x1024@30.0 (4)
4.1	245760	8192	50 Mbit/s	50 Mbit/s	150 Mbit/s	200 Mbit/s	1280x720@68.3 (9) 1920x1088@30.1 (4) 2048x1024@30.0 (4)
4.2	522240	8704	50 Mbit/s	50 Mbit/s	150 Mbit/s	200 Mbit/s	1920x1088@64.0 (4) 2048x1088@60.0 (4)
5	589824	22080	135 Mbit/s	168.75 Mbit/s	405 Mbit/s	540 Mbit/s	1920x1088@72.3 (13) 2048x1024@72.0 (13) 2048x1088@67.8 (12) 2560x1920@30.7 (5) 3680x1536@26.7 (5)
5.1	983040	36864	240 Mbit/s	300 Mbit/s	720 Mbit/s	960 Mbit/s	1920x1088@120.5 (16) 4096x2048@30.0 (5) 4096x2304@26.7 (5)
Level number	Max macroblocks per second	Max frame size (macroblocks)	Max video bit rate (VCL) for Baseline, Extended and Main Profiles	Max video bit rate (VCL) for High Profile	Max video bit rate (VCL) for High 10 Profile	Max video bit rate (VCL) for High 4:2:2 and High 4:4:4 Predictive Profiles	Examples for high resolution @ frame rate (max stored frames) in Level

### 三、其他：

**1.buffer\_size:** 减少卡顿或者花屏现象，相当于增加或扩大了缓冲区，给予编码和发送足够的时间。

例：`av_dict_set(&param, "buffer_size", "1024000", 0);`

**2.rtsp\_transport:** 修改优先连接发送方式，可以用udp、tcp、rtp

例：`av_dict_set(&param, "rtsp_transport", "udp", 0);`

**3.timeout:** 设置超时断开，在进行连接时是阻塞状态，若没有设置超时断开则会一直去阻塞获取数据，单位是微秒。

例：`av_dict_set(&param, "timeout", "5000000", 0);`

**4.movflags:** 加入mp4头。

例：`av_dict_set(&param, "movflags", "empty_moov+default_base_moov+faststart", 0);`

**5.frag\_duration:** 设定mp4容器大小

例：`av_dict_set_int(&param, "frag_duration", 100 * 1000, 0);`

**注：**延迟时间和视频质量一直是一个成反比例的两个参数，牺牲时间还是牺牲质量是在每一个项目中需要根据实际需求去探讨和调整的，但是我们需要做的不仅是对于延迟时间和视频质量比例的调整，更多的还是要利用所拥有的资源去在规定延迟时间内发挥出最好的视频质量。



## FFmpeg帧率设置总结

weixin\_42717961的博客 · 1万+

正常情况下可以通过AVCodecContext结构体中的time\_base来设置帧率，如下设置帧率30 ...

## ffmpeg超时设置

yushanxue的博客 · 4097

1.rtsp超时 AVDictionary\* opts = NULL; av\_dict\_set(&opts, "rtsp\_transport", "tcp"); //设置tc...

## 参与评论

抢沙发



请发表有价值的评论， 博客评论不欢迎灌水，良好的社区氛围需大家一...

评论

## 利用AVDictionary 配置参数\_liujiayu2的专栏

2-21

} AVDictionaryEntry; 下面就用示例的方式简单介绍下用法 (1)创建一个字典 AVDictionary \*d...

## ffmpeg中使用av\_dict\_set接口解决打开rtsp/udp/http等u...

3-3

AVDictionary\* opts = NULL; av\_dict\_set(&opts, "rtsp\_transport", m\_bTcp ? "tcp" : "udp", 0);...

## FFmpeg码率控制及内置编码参数介绍

DONGHONGBAI的专栏 · 1万+

一、码率控制参数 二、编码速度&amp;&amp;编码质量&amp;&amp;视觉优化参数 ...

## FFmpeg类库打开流媒体的方法（需要传参数的时候）

dianyimo9099的博客 · 387

使用ffmpeg类库进行开发的时候，打开流媒体（或本地文件）的函数是avformat\_open\_inp...

## FFmpeg庖丁解牛系列之dict模块\_老王的技术博客

2-25

简言之,就是key:value对存储的。最简单的创建AVDictionary的方式就是给av\_dict\_set()传递...

## FFmpeg源码分析之 av\_opt\_set\_dict()

ice\_ly000的博客 · 1635

1 av\_opt\_set\_dict() av\_opt\_set\_dict() 声明： 所属库： libavutil ( lavu ) 头文件： libavutil/o...

## 如何利用 AVDictionary 配置参数

学而时习之 不亦乐乎 · 8751

本文是我的 FFmpeg Tips 系列的第五篇文章，准备介绍下 ffmpeg 提供的一个非常好用的...

## 记ffmpeg调用av\_dict\_set设置含有"+"号的选项

jfu22的专栏 · 8484

av\_dict\_set(&options, "rtsp\_transport", "+udp+tcp", 0); av\_dict\_set(&options, "rtsp\_flags", "...

## 从学龄前开始解读FFmpeg代码 之 AVDictionary结构体以...

... zzyincsdn的博客 · 536

@[TOC]( 从学龄前开始解读FFmpeg代码 之 AVDictionary结构体以及av\_dict\_set()相关函...

## FFmpeg中AVDictionary介绍 热门推荐

网络资源是无限的 · 6万+

FFmpeg中的AVDictionary是一个结构体，简单的key/value存储，经常使用AVDictionary设...

## ffmpeg设置tcp,连接时间，解码格式

jacke121的专栏 · 5517

pFormatCtx = avformat\_alloc\_context(); AVDictionary\* options = NULL; av\_dict\_set(&optio...

## ffmpeg 码率控制（总结篇）

隔壁老王呀的专栏 · 5684

本文仅探讨ffmpeg API编码的方式进行码率控制，文中测试数据使用的是网络传输场景，非...

## AVDictionary结构体相关源码介绍

华的专栏 · 4449

本文对AVDictionary结构体部分相关函数代码进行了介绍 [cpp] view plain copy 本文研究分...

## ffmpeg使用笔记

奔跑的艾斯的博客 · 3472

记录一下最近对接rtsp摄像头的过程： 1.版本问题 使用2.8.15版本时 pformatContext=avfor...

## 如何通过ffmpeg 实现实时推流和拉流保存的...

最新发布 TSINGSEE官方博客 · 711

FFmpeg是特别强大的专门用于处理音视频的开源库，既可以使用它的API对音视频进行处...

## 【FFmpeg笔记】05-AVDictionary使用介绍

eieihihi的专栏 · 624

1. AVDictionary 介绍 AVDictionary 是一种字典数据结构，可以简单理解为 key-value 集合...

## Qt+ffmpeg实现视频流编码成MP4

程序猿的博客 · 903

1.实现思路 拉流cctv1: http://vi.bupt.edu.cn/hls/cctv1hd.m3u8 1.1编码线程头文件 #pragm...

## ffmpeg硬解码延迟\_关于FFmpeg推流端降低延迟...

weixin\_39887531的博客 · 655

本文所写的是调节推流端降低延迟能调节的一些方案，有不足请补充，其他端方向类似，例...

## ffmpeg接收媒体流时的设置--转

洛的博客 · 922

AVDictionary\* options = NULL; av\_dict\_set(&options, "buffer\_size", "1024000", 0); //增大"b...

“相关推荐”对你有帮助么？



非常没帮助



没帮助



一般



有帮助



非常有帮助





ZackZheng999

码龄3年 暂无认证

13	84万+	13万+	2万+	
原创	周排名	总排名	访问	等级
328	161	14	14	72
积分	粉丝	获赞	评论	收藏



私信

关注

搜博主文章



### 热门文章

Jupyter中显示数据data时只显示省略号不显示完整数据 5873

Jupyter 中 ValueError: Duplicate names are not allowed.的问题解决 4260

ffmpeg推流 av\_dict\_set 参数设置解析（降低延时、处理花屏、改善画面质量）（实时更新） 2662

MFC 程序大小及控件自适应不同的屏幕分辨率 2272

ffmpeg 推流 在H264中插入SPS、PPS 头 1046

### 分类专栏

	Linux	1篇
	ffmpeg	5篇
	C++	5篇
	rtsp	1篇
	pcm	1篇
	SDL2	1篇



### 最新评论

ffmpeg filter 实现画面旋转  
初见-Android: 大佬您好，感谢您的回复，问题已经解决啦编码器应该在使用filter时...

ffmpeg filter 实现画面旋转  
ZackZheng999: 我这个因为做的是播放，所以是在解码之后进行旋转。如果你要...

ffmpeg filter 实现画面旋转  
ZackZheng999: 这个是解码出来之后拿yuv数据进行filter，在avcodec\_send\_frame ...

ffmpeg filter 实现画面旋转



ZackZheng999

关注

0

ffmpeg filter 实现画面旋转

莫得感情的bug制造工程师: 写得好！膜拜大神！

您愿意向朋友推荐“博客详情页”吗？



强烈不推荐



不推荐



一般般



推荐



强烈推荐

#### 最新文章

ffmpeg filter 实现画面旋转

windows下编译ffmpeg生成dll、lib库（开启265）

windows API waveout 双缓冲区播放pcm实时语音流

2021年 7篇

2020年 7篇

#### 目录

一、关于影响时延的参数设置

二、关于影响视频质量的参数设置

三、其他：

注：延迟时间和视频质量一直是一个成反...



ZackZheng999

关注

👍 0

