

原创文三于 2020-11-27 15:53:35 发布2302收藏7

分类专栏:ffmpeg

ffmpeg 专栏收录该内容3 订阅40 篇文章订阅专栏

对于一个h264码流，若想要改变帧间隔，需要解码获取原始数据再针对原始数据进行编码。
此代码把帧间隔为25的h264码流改变成帧间隔20。

ffmpeg 版本4.3

```
1 static void openEncoder(int width, int height, AVCodecContext** enc_ctx)
2 {
3     //使用Libx264编码器
4     AVCodec * pCodec = avcodec_find_encoder_by_name("libx264");
5     if(nullptr == pCodec)
6     {
7         printf("avcodec_find_encoder_by_name fail.\n");
8         return;
9     }
10    //获取编码上下文
11    *enc_ctx = avcodec_alloc_context3(pCodec);
12    if(nullptr == enc_ctx)
13    {
14        printf("avcodec_alloc_context3(pCodec) fail.\n");
15        return;
16    }
17    //sps/pps
18    (*enc_ctx)->profile = FF_PROFILE_H264_HIGH;
19    (*enc_ctx)->level = 50;//表示Level是5.0
20    //分辨率
21    (*enc_ctx)->width = width;
22    (*enc_ctx)->height = height;
23    //gop
24    (*enc_ctx)->gop_size = 20;
25    (*enc_ctx)->keyint_min = 15;//设置最小自动插入I帧的间隔.OPTION
26    //B帧
27    (*enc_ctx)->max_b_frames = 0;//OPTION
28    (*enc_ctx)->has_b_frames = 0;//OPTION
29    //参考帧
30    (*enc_ctx)->refs = 3;//OPTION
31    //设置输入的yuv格式
32    (*enc_ctx)->pix_fmt = AV_PIX_FMT_YUV420P;
33    //设置码率
34    (*enc_ctx)->bit_rate = 1000000;
35    //设置帧率
36    (*enc_ctx)->time_base = (AVRational){1,25};//帧与帧之间的间隔
37    (*enc_ctx)->framerate = (AVRational){25,1};//帧率 25帧每秒
38    if(avcodec_open2(*enc_ctx,pCodec,nullptr) < 0)
39    {
40        printf("avcodec_open2 fail.\n");
41    }
42    return;
43 }
44
45 static AVFrame* creat_frame(int width,int height)
46 {
47     AVFrame *frame = av_frame_alloc();
48     frame->width = width;
49     frame->height = height;
50     frame->format = AV_PIX_FMT_YUV420P;
51     //分配frame中buffer的大小
52     av_frame_get_buffer(frame, 32);//按32位对齐
53     return frame;
54 }
55
56 static void encode(AVCodecContext *encCtx,AVFrame *frame,AVPacket *newpkt,FILE *outPutFile)
57 {
58     int nRet = 0;
59     if (frame)
60         printf("send frame to encoder ,pts=%lld \n",frame->pts);
61     //送原始数据给编码器进行编码
62     nRet = avcodec_send_frame(encCtx,frame);
63     if(nRet < 0)
64     {
65         printf("avcodec_send_frame fail.\n");
66         return;
67     }
68     //从编码器获取编号的数据
69     while(nRet >= 0)
70     {
71         nRet = avcodec_receive_packet(encCtx,newpkt);
72         if(nRet < 0)
73         {
74             printf("avcodec_receive_packet fail.\n");
75             return;
76         }
77         fwrite(newpkt->data,1,newpkt->size,outPutFile);
78         av_packet_unref(newpkt);
79     }
80 }
81 #define YUVFORMAT_YUV420P AV_PIX_FMT_YUV420P
82 static void test()
83 {
84
85     int nRet = 0;
86     const char *pInFileName = "D:/videos/gop30.h264";
87     const char *pOutFileName = "D:/output/gop20.h264";
88     FILE *fp = fopen(pOutFileName, "wb+");
89     AVDictionary *pDic = nullptr;
90     AVFormatContext *pInFmtCtx = nullptr;
91     nRet = avformat_open_input(&pInFmtCtx,pInFileName,nullptr,&pDic);
92     if( nRet < 0)
93     {
94         printf("Could not open input file.");
95         return;
96     }
97     avformat_find_stream_info(pInFmtCtx, nullptr);
98     printf("*****Input Information*****\n");
99     av_dump_format(pInFmtCtx, 0, pInFileName, 0);
100    printf("*****\n");
101    //查找编码器
102    int vudioStreamIndex = -1;
103    for(int i = 0; i < pInFmtCtx->nb_streams; ++i)
104    {
105        if(AVMEDIA_TYPE_VIDEO == pInFmtCtx->streams[i]->codecpar->codec_type)
106        {
107            vudioStreamIndex = i;
```

分类专栏

笔记	5篇
C++	10篇
ffmpeg	40篇
stl	
VS	5篇
H265	
QT5	19篇
SDL2	3篇
QT+FFMPEG	
TS	
日常记录	2篇
ubuntu	2篇
QT for Andriod	1篇
数字图像处理-冈萨雷斯	
andriod	1篇
opengl	
android studio	
UOS	1篇
H264	
音频	
QT for Linux	1篇
onvif	2篇
msys2	1篇
QT+SDL	2篇

```
208 break;
209 }
210 }
211 AVStream * in_stream = pInFmtCtx->streams[vudioStreamIndex];
212 AVCodec *pInCodec = avcodec_find_decoder(in_stream->codecpar->codec_id);
213 if(nullptr == pInCodec)
214 {
215     printf("avcodec_find_decoder fail.");
216     return;
217 }
218 AVCodecContext* pInCodecCtx = avcodec_alloc_context3(pInCodec); //????????????
219 nRet = avcodec_parameters_to_context(pInCodecCtx, in_stream->codecpar);
220 if(nRet < 0)
221 {
222     printf("avcodec_parameters_to_context fail.");
223     return;
224 }
225 //打开解码器
226 if(avcodec_open2(pInCodecCtx, pInCodec, nullptr) < 0)
227 {
228     printf("Error: Can't open codec!\n");
229     return ;
230 }
231 printf("width = %d\n", pInCodecCtx->width);
232 printf("height = %d\n", pInCodecCtx->height);
233 AVFrame *pFrameYUV = av_frame_alloc();
234 AVPacket *packet = av_packet_alloc();
235 av_init_packet(packet);
236 //计算这个格式的图片，需要多少字节来存储
237 int bytes_num = av_image_get_buffer_size(YUVFORMAT_YUV420P, pInCodecCtx->width, pInCodecCtx->height,1);
238 //申请空间来存放图片数据，包含源数据和目标数据
239 uint8_t* out_buffer = (uint8_t*)av_malloc(bytes_num);
240 //前面的av_frame_alloc函数，只是为这个AVFrame结构体分配了内存，
241 //而读类型的指针指向的内存还没分配，这里把av_malloc得到的内存和AVFrame关联起来。
242 //当然，其还会设置AVFrame的其他成员
243 avpicture_fill((AVPicture*)pFrameYUV, out_buffer, YUVFORMAT_YUV420P,pInCodecCtx->width, pInCodecCtx->height
244 int got_picture = 0;
245 AVFrame *pFrame = av_frame_alloc();
246 AVPacket *newpkt = av_packet_alloc();
247 int64_t pts = 0;
248 AVCodecContext *encCtx = nullptr;
249 //打开编码器
250 openEncoder(1920,1080,0encCtx);
251 while(av_read_frame(pInFmtCtx, packet) >= 0)
252 {
253     if( vudioStreamIndex == packet->stream_index)
254     {
255         //avcodec_send_packet送原始数据给编码器进行编码
256         //avcodec_receive_frame
257         if(avcodec_send_packet(pInCodecCtx, packet)<0 || (got_picture =avcodec_receive_frame(pInCodecCtx, p
258         {
259             std::cout<<"changeGOP end";
260             goto __end;
261         }
262         if(!got_picture)//
263         {
264             pts += 40;
265             // 设置pts
266             pFrame->pts = pts;
267             encode(encCtx, pFrame, newpkt, fp);
268         }
269         av_packet_unref(packet);
270         av_packet_unref(newpkt);
271     }
272 }
273 fflush(fp);
274 //flush decoder
275 //当av_read_frame 退出循环的时候，实际上解码器中可能还包含
276 //剩余的几帧数据，直接调用avcodec_decode_video2获得AVFrame ,
277 //而不再向解码器传递AVPacket
278 while(1)
279 {
280     if(avcodec_send_packet(pInCodecCtx, packet)<0 || (got_picture =avcodec_receive_frame(pInCodecCtx, pFram
281     {
282         std::cout<<"changeGOP end";
283         goto __end;
284     }
285     if(!got_picture)//
286     {
287         pts += 40;
288         // 设置pts
289         pFrame->pts = pts;
290         encode(encCtx, pFrame, newpkt, fp);
291     }
292     av_packet_unref(packet);
293     av_packet_unref(newpkt);
294 }
295 __end:
296     fflush(fp);
297     fclose(fp);
298     av_frame_free(&pFrame);
299     av_frame_free(&pFrameYUV);
300     avcodec_close(pInCodecCtx);
301     avformat_close_input(&pInFmtCtx);
302 }
303
304
305 void main()
306 {
307     test();
308     std::cout<<"changeGOP end";
309 }
```

利用ffmpeg将H264裸流文件封装为MP4文件并保存

06-29

利用ffmpeg库，将H264裸流文件读到内存，然后封装为MP4文件并保存到本地。转换效果亲测，流畅，无卡顿。

ffmpeg设置h264编码IDR间隔

zhuxian2009的专栏 2728

在ffmpeg中，编码h264时，想强制刷出一个帧，比较容易，但是想强制刷出IDR帧，却没有明确的说明。而在x264的头文件中，可以看到一个参数说明：...

使用ffmpeg时，如何设置恒定码率并控制好关键帧间隔

7-18

(1)-pass 1 -passlogfile ffmpeg2pass 第一步先编一次生成 ffmpeg2pass 文件 (2)-pass 2 -passlogfile ffmpeg2pass 第二次会根据第一次生成的ffmpeg2pa...

ffmpeg_ffmpeg 帧率_qq_42331496的博客

7-29

ffmpeg -re -stream_loop -1 -flags +genpts -i c:\aa\aa.mp4 -vcodec copy -acodec copy -strict -2 -f flv -y rtmp://localhost:1935/hls1/test03 4 调节帧率 帧...

MediaCodec 解码H264/H265码流视频

m0_60259116的博客 190

MediaCodec解码H264/H265码流视频

使用ffmpeg将多张图片生成H264裸流并获取帧图片

10-10

1.使用ffmpeg将多张图片生成H264文件； 2.从H264文件中获取一帧图片和一帧中的SLICE图片； 3.能够直接使用mplay.exe播放H264文件； 4.完整工程，...

使用ffmpeg、x264库强制编出关键帧(帧) ffmpeg i帧 太上绝情的博客-CSD...

7-27

使用ffmpeg、x264库强制编出关键帧(帧) 背景 当h264数据在使用udp传输时，丢包是一件很常见的事情，当接收端丢包发生后，应该通知发送端，让发送端立...



文三-

码龄6年

暂无认证

88

6万+

194万+

12万+



原创

周排名

总排名

访问

等级

1241

27

83

54

398

积分

粉丝

获赞

评论

收藏



私信

关注

搜博文文章

热门文章

ffmpeg通过rtsp服务器推流拉流--命令行实现

39676

Two slices reporting being the first in the same frame. 和 Could not find ref with POCXX问题

3882

windows使用QT Creator开发的QT程序移植到Ubuntu18, 以及打包发布

3876

记录一下QT movetothread多线程槽函数无法触发

3807

ffmpeg使用qsv解码遇到的问题

2640

最新评论

ffmpeg使用qsv解码遇到的问题

shengutong9765: 请问qsv 解码的延迟很高, 再显示的延迟很高 是又解码缓冲区 参...

VS+QT开发Ocx/ActiveX控件 -- m0_74362852: twzasbbotfz

QT on Android的rtsp播放器demo liuzizhuo123: 求源码, 老哥

ffmpeg通过rtsp服务器推流拉流--命令行... m0_60535482: 链接不到rtsp网址

ffmpeg H264转YUV420p

嘿嘿我是水上麻鹿: [h264 @ 0000000002a15c40] decoding for stream 0 failed 请...

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

windows pcapng文件抽出udp负载

英伟达GPU 解码&编码 能力

windows下WS2tcpip.h和windows.h报错

2022年 13篇

2021年 45篇

2020年 36篇

使用ffmpeg处理 视频中帧间时间戳异常、关键帧间隔异常_weixin_34...

8-1

1、视频中帧间时间戳异常(理论两帧差在40毫秒左右,V网通的有11777毫秒的情况) 2、关键帧间隔异常(发现出现了2分钟才出现关键帧的情况.正常2~...

ffmpeg设置视频帧率 热门推荐

se00_2012的专栏

6万+

使用-r参数可以对视频的帧率进行设置, 那么是写成ffmpeg -i input -r n output还是ffmpeg -r n -i input output呢? 实际上-r参数用来设置输入文件和输出文...

FFMPEG命令 FFMPEG命令粗略介绍

SanShuiGeGe的博客

865

ffmpeg推流时循环播放不中断, 强制关键帧间隔为1秒_路边闲人2的博客-C...

7-19

下面这个超级准确: 每1秒钟一个关键帧. jobs/ffmpeg/bin/ffmpeg -re -stream_loop -1 -i ./doc/jqr.mkv \ -vcodec libx264 -g 25 -x264-params keyint=25n...

使用ffmpeg编码时, 如何设置恒定码率, 并控制好关键帧帧间隔

SanShuiGeGe的博客

3877

使用ffmpeg编码时, 如何设置恒定码率, 并控制好关键帧帧间隔

视频h264格式文件可行的改变帧率FPS的命令ffmpeg

u010879745的博客

2485

经本人亲试, 可行命令如下: 转成mp4时, 用不用-vcodec copy均可, 不用时处理时间长一些 ffmpeg -r 2 -f h264 -i city.h264 -vcodec copy -an output2....

ffmpeg强制关键帧间隔(key frame, gop size, gop duration)

qq_29007291的博客

2555

ffmpeg改变视频的GOP size

H264编码器参数理解 (部分) 1帧和帧间距等的理解

java开发指南博客【转载】

1005

写作背景: 最近在研究H264编码器的参数, 里面的参数不太懂.其中包括: 1) 30# i intervals // 1 帧间距 2) 30# idr intervals// idr帧间距 3) 0# b frame nu...

ffmpeg常用命令

terrence_hsu的博客

2773

ffmpeg 常用参数 ifprobe 常用参数

ffmpeg解码MP4生成h264和Opus裸流数据

12-05

ffmpeg解码MP4生成h264和Opus裸流数据, 不只是MP4.其他的封装格式也行, 只要ffmpeg支持的就可以。由于ffmpeg不支持获取opus的裸流数据, 输出...

FFmpegTool_ffmpeg_h264编码_ffmpeg_h264H265解码编码_源码

10-04

主要实现H264和H265的解码和编码, 运用FFMPEG核心来处理。

使用ffmpeg封装h264裸流为MP4文件

11-21

使用ffmpeg封装h264裸流为MP4文件, 亲测, 可修改时间戳实现输出视频的速度控制, 附使用说明。

python+ffmpeg 获取rtsp码流的帧间隔

es15071848238的博客

3352

测试的小伙伴们应该了解过, 获取帧间隔的方法: 1. 录制H264码流, 然后使用雷神的H264分析工具来查看帧间隔, 这个方法比较直观, 可以查看详细的...

【FFmpeg】设置H264参数

郭老二

5331

一、设置x264参数的接口 // 获取编码器 AVCodec *codec = avcodec_find_encoder(AV_CODEC_ID_H264); // 创建编码器上下文 AVCodecContext *codecx...

ffmpeg每隔几帧(间隔) 提取图片的方法

大数据挖掘SparkExpert的博客

1万+

在一些视频分类任务中, 往往需要从视频中提取指定帧, 提取 R G B 信息然后进行训练和分类。提取帧的方法有很多, 不过在一些对时间限制比较大的场...

FFmpeg获取H264文件的帧率

Griza_J的博客

4098

开发中常常需要确定一个H264文件的帧率信息, 在AVFormatContext的视频流stream中记录着两种帧率avg_frame_rate(平均帧率), r_frame_rate(基本帧率)。

ffmpeg 编码h264 最新发布

07-22

你可以使用FFmpeg来编码H.264视频。下面是一个示例命令: `` ffmpeg -i input.mp4 -c:v libx264 output.mp4 `` 这个命令将输入文件input.mp4编码为H...

“相关推荐”对你有帮助?

非常没帮助

没帮助

一般

有帮助

非常有帮助

关于我们 招贤纳士 商务合作 寻求报道

400-660-0108 kefu@csdn.net 在线客服 工作时间 8:30-22:00

文三

关注

1 7 0

专栏目录

Beta

举报