# XFFMpegCodec编码参数说明

V 1.0

文档修订记录

|  |  |  |  |
| --- | --- | --- | --- |
| **版本** | **修订日期** | **修订人** | **修订摘要** |
| V1.0 | 2017-12-19 | 骆新 | 初稿 |
|  |  |  |  |

## 概述

XFFMpegCodec是一个封装了ffmpeg编解码功能的动态库。常规的解码过程相对简单，需要传递给解码器的参数较少，而编码过程需要传入的参数相对较多。在这些编码参数中，有一部分公共基础参数可以通过统一的C++结构传递给编码器，但还有一部分属于是编码格式的个性化参数，无法通过统一而且简单的C++结构传给编码器。

在XFFMpegCodec的编码中，通过额外字符串lpszExtParam传入编码个性参数。该字符串统一使用UTF-8编码，key=value的形势，可以有多个参数，参数间用‘；’隔开。例如：

“encoder=libx264;preset= ultrafast”

另外，每种编码格式对于公共参数的适用范围也是有要求的，本文档也会进行相应说明。

## 公共编码参数说明

### 视频公共编码参数

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **说明** |
| cf | ColorFormat | 编码的颜色格式 |
| szVideo | SIZE | 视频画面的大小 |
| rtDAR | XRate | 显示比例 |
| vsm | VideoScanMode | 扫描方式 |
| bIsClipped | bool | 颜色值的范围（16-235 or 0-255） |
| clrPrim | ColorPrimaries | 源基色的色度坐标(ISO13818-2) |
| clrTrans | ColorTransfer | 光电转换特性(ISO13818-2) |
| clrSpace | ColorSpace | 颜色空间变换系数(ISO13818-2) |
| chrLoct | ChromaLocation | Location of chroma samples |
| afd | ActFmtDesc | 图像有效区域信息描述，按SMPTE 2016-1定义 |
| bVarFPS | bool | 是否为变帧率 |
| rtFPS | XRate | 帧率 |
| codecType | XVideoCodecType | 编码格式 |
| codecTag | DWORD | 编码格式子类型，fourcc |
| nGopSize | long | Gop大小 |
| nBitRate | long | 码率 bit/s |

### 音频公共编码参数

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **说明** |
| Channels | short | 通道数量 |
| SampPerSec | long | 采样率 |
| res | WORD | ARes，枚举类型的量化精度 |
| chLayout | LXCHLayout | 通道排列方式 |
| bPlanar | bool | 通道数据排列方式，P排列，或者是打包方式 |
| codecType | XVideoCodecType | 编码格式 |
| codecTag | DWORD | 编码格式子类型，fourcc |
| nBitRate | long | 码率 bit/s |

## 视频编码格式说明

### GIF编码说明

#### 颜色格式

目前只实现了PAL8形式的图像编码。公共参数中的颜色格式，无论指定为何都必须传递PAL8图像进行编码。

有关PAL8格式的说明见附件“PAL8格式说明”。

#### 透明GIF编码

编码器会将第一帧图像的索引表中A值最小的那个索引色作为透明色。

目前对于透明GIF编码实现存在问题，多帧图像会出现叠加，因此还不能使用。

#### XCodec中的GIF编码器

需要进行GIF编码时建议使用XCodec中封装好的GIF编码器。

HRESULT CreateGifEncoder(OUT IXGifEncoder\*\* ppEncoder);

它会对传入的BGRA或者BGR24图像进行处理生成PAL8格式图像。

### h264编码参数

普通方式进行h264编码时采用的是libx264编码器。

#### lpszExtParam参数

**encoder：**编码器名称不用特别指定

**preset：**预设

* + - veryslow
    - slower
    - slow
    - medium
    - fast
    - faster
    - veryfast
    - superfast
    - ultrafast

#### 颜色格式

* + YUV420P，YUVJ420P
  + YUV422P，YUVJ422P
  + YUV444P，YUVJ444P
  + NV12，NV21
  + NV16

### nvenc h264编码参数

使用NVIDIA显卡进行h264编码

#### lpszExtParam参数

**encoder：**编码器名称，下面任选其一

**nvenc / nvenc\_h264 / h264\_nvenc**

**preset：**预设

* + - slow ：hq 2 passes
    - medium ：hq 1 pass。(default – 默认值或者指定为default时采用)
    - fast ：hp 1 pass
    - hp
    - hq
    - bd
    - ll ：low latency
    - llhq ：low latency hq
    - llhp ：low latency hp
    - lossless
    - losslesshp

#### 颜色格式

* + YUV420P
  + NV12
  + YUV444P
  + YUV444P16LE
  + BGRX，RGBX
  + CUDA，D3D11

### nvenc hevc编码参数

使用NVIDIA显卡进行hevc编码

#### lpszExtParam参数

**encoder：**编码器名称，下面任选其一

**nvenc\_hevc / hevc\_nvenc**

**preset：**预设

* + - slow ：hq 2 passes
    - medium ：hq 1 pass。(default – 默认值或者指定为default时采用)
    - fast ：hp 1 pass
    - hp
    - hq
    - bd
    - ll ：low latency
    - llhq ：low latency hq
    - llhp ：low latency hp
    - lossless
    - losslesshp

#### 颜色格式

* + YUV420P
  + NV12
  + YUV444P
  + YUV444P16LE
  + BGRX，RGBX
  + CUDA，D3D11

### qsv h264编码参数

使用intel核显进行h264编码

#### lpszExtParam参数

**encoder：**编码器名称

**h264\_qsv**

**preset：**预设

* + - veryfast
    - faster
    - fast
    - medium
    - slow
    - slower
    - veryslow

#### 颜色格式

* + NV12

### qsv hevc编码参数

使用intel核显进行hevc编码

#### lpszExtParam参数

**encoder：**编码器名称

**hevc\_qsv**

**preset：**预设

* + - veryfast
    - faster
    - fast
    - medium
    - slow
    - slower
    - veryslow

#### 颜色格式

* + NV12

### qsv mpeg2编码参数

使用intel核显进行mpeg2编码

#### lpszExtParam参数

**encoder：**编码器名称

**mpeg2\_qsv**

**preset：**预设

* + - veryfast
    - faster
    - fast
    - medium
    - slow
    - slower
    - veryslow

#### 颜色格式

* + NV12

## 音频编码格式说明

### AAC

### MP3

## 附件

### PAL8格式说明

PAL8是一种最多支持256种颜色的索引颜色格式。它带有一个256单元，大小为1024字节的颜色查找表。每个单元都是一个BGRA颜色，各分量按BGRA顺序从低到高字节排列，各占8bit。其中A通常无定义，建议设置为255。图像部分，每个像素占用1个字节，值为查找表单元的索引。

XFFMpegCodec编码器的图像传递采用下面的结构。

struct SVideoBuf

{

PBYTE data[4]; //记录各个视频分量的地址

int linesize[4];

};

data[0] 指向图像部分

data[1] 指向颜色查找表

linesize[0] 为图像一行的跨距