

# FRDS新手压制流程指南

## 前言

越来越多新人加入了压制行列，但新人上手不熟悉流程，对教程不理解，容易踩到许多坑；且大佬们不是24小时在线，新人可能遇到的坑是前人踩过，反复回复相同问题增加了大佬的负担。故写下此指南，进一步降低教程门槛，使新人更易做鸭上手。

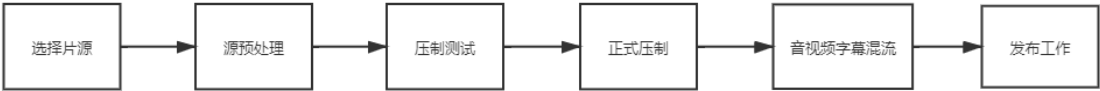
**必须阅读一遍[官方教程](#)了解原理，再阅读本指南，可能对整个压制流程能有一个大致的认识。**

本文因编者水平有限，仅作为新人上手指导，其中不乏有疏漏，还望各位大佬指正，若有本文中没有提到的情况或其他不解之处，请在咨询大佬，在此对大佬的无私指导和指正表示**感谢**！

指南源格式为Markdown，为发布方便输出为PDF格式，可能观感与复制操作困难，敬请谅解。

## 目录

常规压制流程



### FRDS新手压制流程指南

前言

目录

Let's Do This!

环境配置

选择片源

源预处理

原盘信息判断

预混流

压制测试

代码小课堂

压制脚本调整

压制参数配置

正式压制

音视频字幕混流

音频处理

字幕获取

混流

发布工作

结语

引用

更新日志

## Let's Do This!

本文主要讲解 Windows10/11 x64 下压制环境的配置与使用，其他环境的用户可以选择性阅读。

Linux 环境在此推荐[HLW组御用Docker](#)；MacOS 环境配置起来极其困难，M1 Series不适用于配置压制环境，建议放弃。

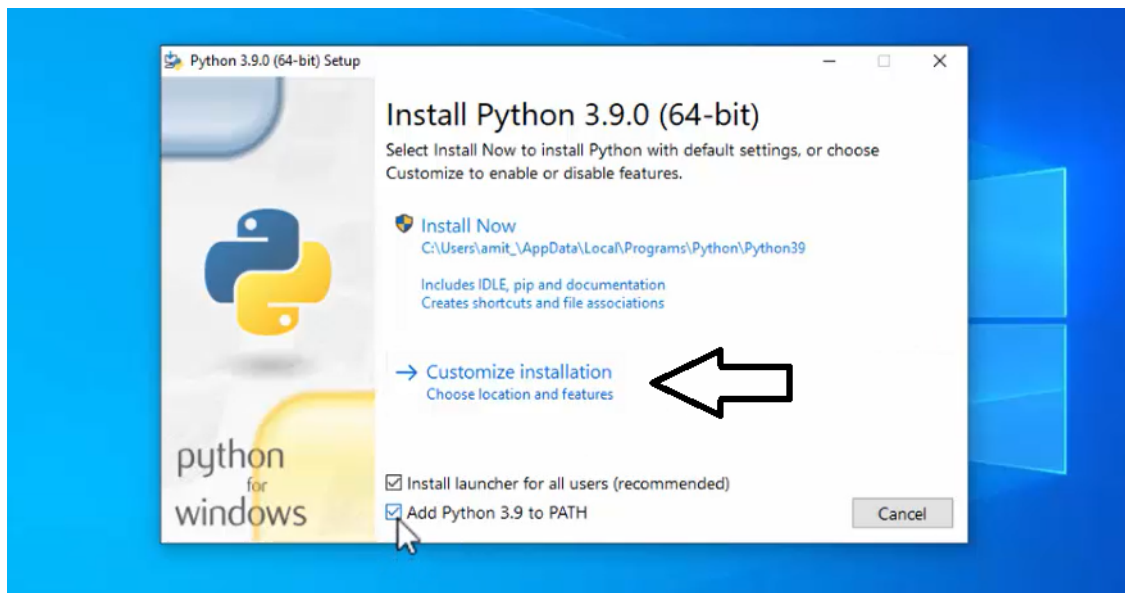
## 环境配置

- [压制部署包](#)

压制环境及工具打包，可以直接下载解压使用，省去了新人寻找压制工具的烦恼。有能力的可以手动部署相关工具。

- [Python 3.9.x](#)

Python环境，建议安装，注意勾选 **Add Python 3.9 PATH**



或者可以选择不安装，使用压制部署包内附带的Python，并手动将以下两个路径添加至[环境变量](#)。

```
MediaTool\x264_launcher\extra\VapourSynth-64\  
MediaTool\x264_launcher\extra\VapourSynth-64\Scripts\
```

- [Mediainfo/PotPlayer/k-lite](#)

用于生成媒体文件的信息。

- (可选) [DDP\(EAC3\)制作用虚拟机包](#)

用于将DTS/TrueHD等无损音频进行转换的工具。

仅可以在MacOS High Sierra/Mojave/Catalina下运行，搭建环境教程请另寻。

- (可选) [BDInfo](#)

用于查看蓝光原盘中的mpls信息。

- (可选) PowerDVD/DVDFab Player 5/[VLC](#)

用于原盘的观看和查看菜单信息，比较推荐DVDFab Player 5，大部分DIY原盘都可以用其打开。

- (可选) [DGremux](#)

用于原盘的内容提取，类似于Eac3to但对[音频时间戳友好](#)，避免可能出现的音画不同步问题。

- (可选) [SubtitleEdit/Aegisub](#)

用于字幕的查看和编辑。

- (可选) [noMacs](#)

用于切换查看对比截图。

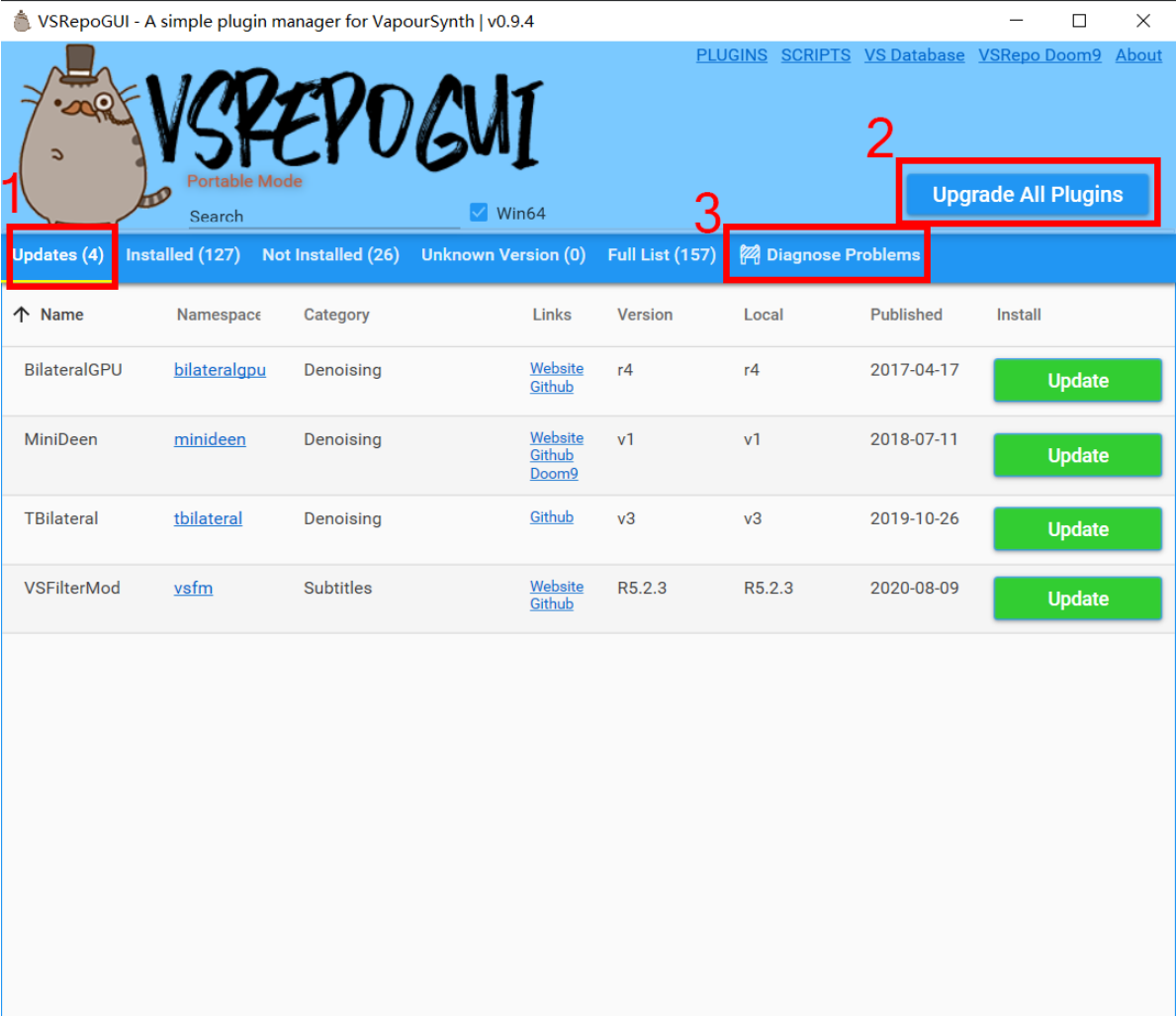
- (可选) [ChapterTool](#)

用于在原盘中没有章节信息时手动生成。

以上工具解压(路径请勿带中文)/安装、配置后，建议使用脚本生成快捷方式方便寻找各软件。双击运行CreatLnk.bat，所有工具的快捷方式将会在\_Lnk中生成。

开始之前，需要对VapourSynth的插件进行升级，打开\_Lnk下的VSRepoGUI，按步骤更新插件并检查。

如果在使用其他软件时提示更新，请更新！



## 选择片源

这里引用基础教程的话语。

要想做一个较为高质量作品，应采用最好的来源进行压制。HDR电影来源较为单一，4k的原盘和Remux资源为主，版本较少能选择不多。SDR目前可用压制版本较多，各个发行商的在不同时期也发行过不同蓝光版本，随着web的兴起，AZ和NF也都发布了较高码率的4k的sdr版本，所以在SDR压制时，需要对于来源进行对比，选取最为高质量的来源进行压制。对于蓝光原盘与remux两者相同时候，推荐使用Remux的更为方便。

总的说来，建议选择电影1080p原盘或Remux，新上手不推荐压制电视剧和演唱会或4K分辨率电影，电视剧大多为WEB-DL，不便确定其源质量；演唱会可能会遇到去交错、音源选择甚至字幕问题，略有复杂不便上手。

DIY盘中可能会配有中文字幕，少去寻找字幕的步骤。

选定资源前请务必确认是否有禁转、禁止二次压制等提示语！

# 源预处理

## 原盘信息判断

- 从种子页面判断原盘中主要部分

对于原盘来说，发布时BDInfo是必须的（可以认为是原盘的mediainfo）。找到**PLAYLIST REPORT**下的**Name**，即为目标MPLS。

某些原盘因其版本原因，可能在BDInfo内展示多个MPLS，请按需选择。

引用

DISC INFO:

Disc Title: No.Time.to.Die.2021.BluRay.1080p.AVC.Atmos.TrueHD7.1-MTeam  
Disc Size: 46,494,765,537 bytes  
Protection: AAC5  
Extras: BD-Java  
BDInfo: 0.7.5.9 (compatible layout created by DVDFab 11.1.0.7)

PLAYLIST REPORT:

Name: 00612.MPLS  
Length: 2:43:24.795 (h:m:s.ms)  
Size: 45,821,233,152 bytes  
Total Bitrate: 37.39 Mbps

VIDEO:

Codec	Bitrate	Description
MPEG-4 AVC Video	29395 kbps	1080p / 23.976 fps / 16:9 / High Profile 4.1

AUDIO:

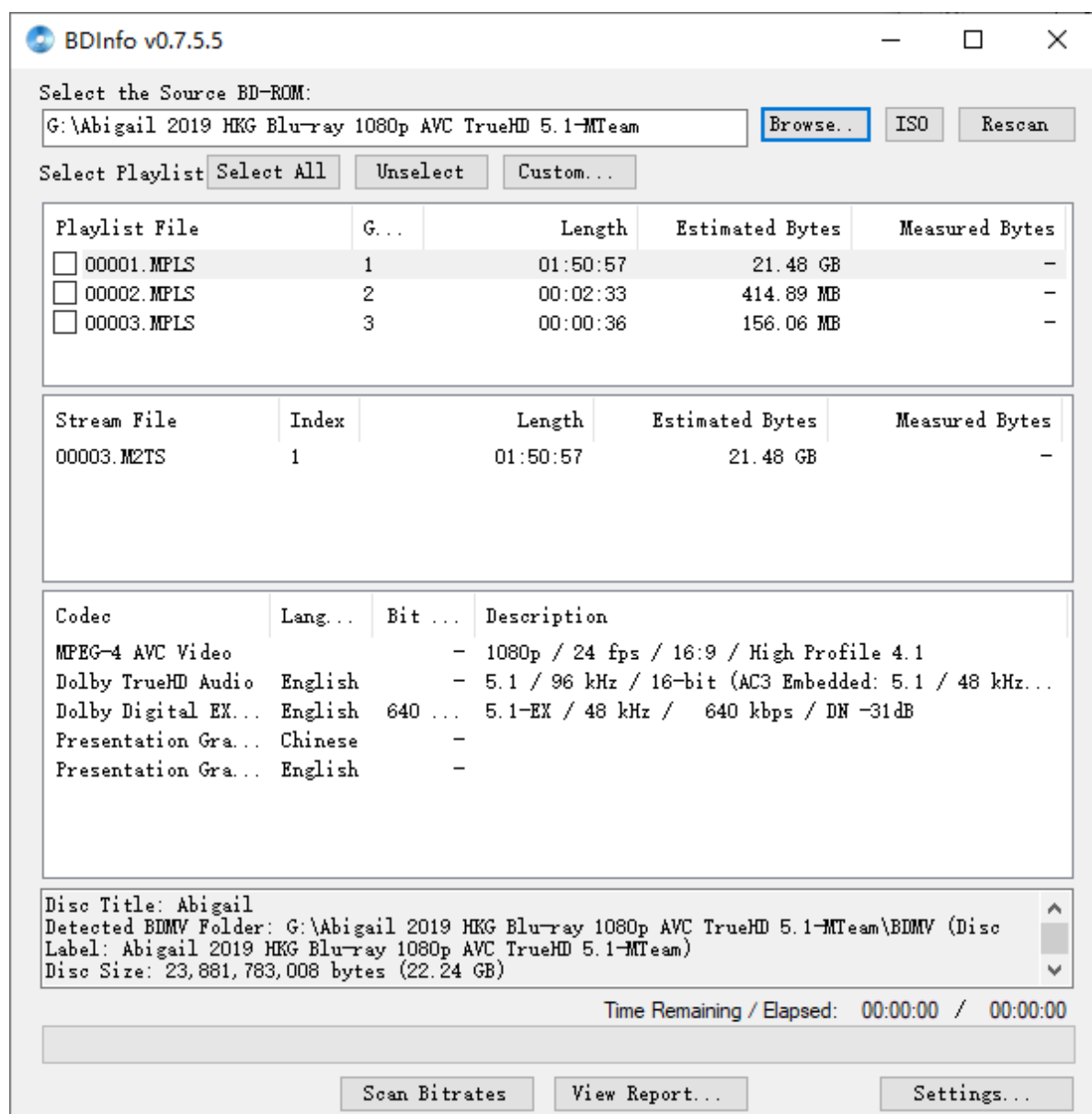
Codec	Language	Bitrate	Description
Dolby TrueHD/Atmos Audio	English	3508 kbps	7.1+11 objects / 48 kHz / 3060 kbps / 16-bit (AC3 Core: 5.1 / 48 kHz / 448 kbps)
Dolby Digital Plus Audio	Spanish	768 kbps	7.1 / 48 kHz / 768 kbps (AC3 Core: 5.1 / 48 kHz / 448 kbps ( Core: 5.1 / 48 kHz / 448 kbps))
Dolby Digital Plus Audio	French	768 kbps	7.1 / 48 kHz / 768 kbps (AC3 Core: 5.1 / 48 kHz / 448 kbps ( Core: 5.1 / 48 kHz / 448 kbps))
Dolby Digital Audio	English	192 kbps	2.0 / 48 kHz / 192 kbps

SUBTITLES:

Codec	Language	Bitrate	Description
Presentation Graphics	English	34.081 kbps	
Presentation Graphics	Spanish	27.218 kbps	
Presentation Graphics	French	27.468 kbps	
Presentation Graphics	Spanish	0.063 kbps	
Presentation Graphics	French	0.262 kbps	

- 使用BDInfo判断原盘中主要部分

对于原盘使用BDInfo检查原盘信息，查看原盘主要视频对应播放列表。对于复杂原盘，可能出现多版本混合的情况，需要确定所需要版本对应的MPLS，并确定原盘的主要码率。



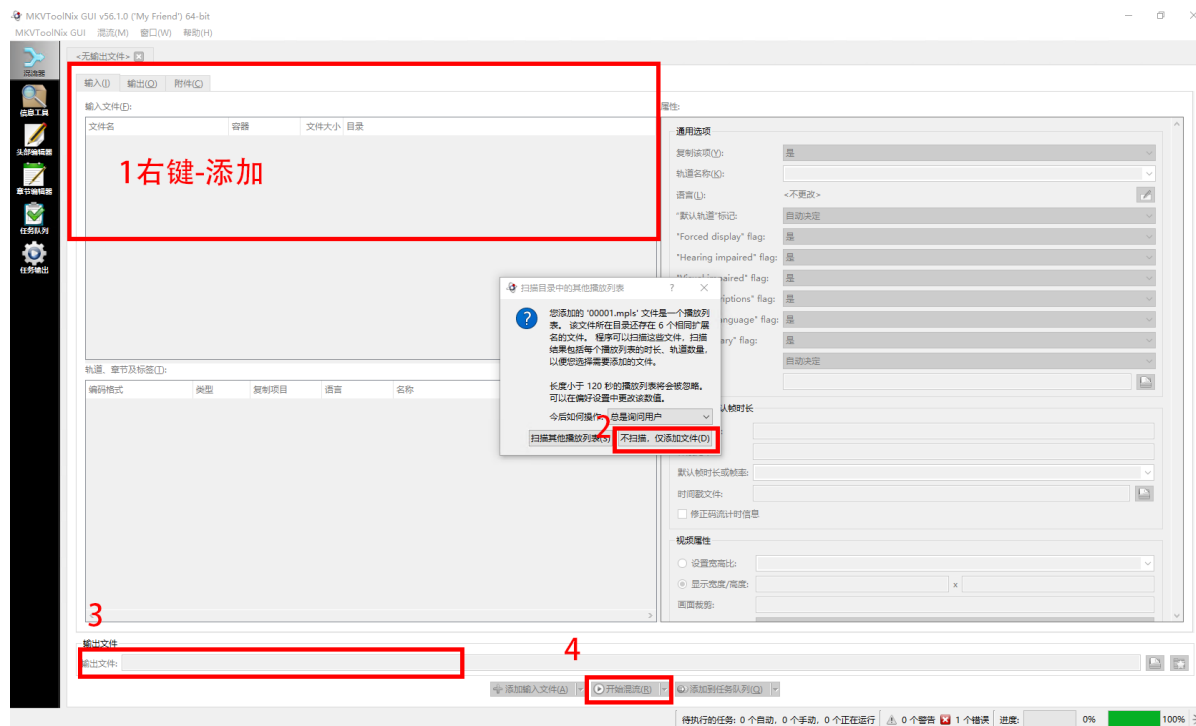
一般时长最长且文件最大的 .mpls 为目标播放列表，记下其名称，后续会用到。本文以 00001.mpls 举例。

## 预混流

使用 *eac3to/DGremux* 提取视频流的可以忽略此步骤。

使用 *MKVtoolnix* 将原盘进行混流，打包为一个完整的影片(可以仅选择视频以节省体积)，方便作为VS的输入源。同时该操作有效防止肉酱盘中多m2ts文件致使压制输入源不便。

右键输入处空白-添加文件，选择之前确定的 00001.mpls，随后的提示框选择"不扫描，仅添加文件"，命名输出，本文将 frdsnb.mkv 举例，点击开始混流。



## 压制测试

以下均以**VapourSynth Editor**为使用范例，若想使用AVS请回看[官方教程](#)。

插入一段零基础代码小课堂，有Python基础的同学可以跳过。

### 代码小课堂

**注释就是对代码的解释和说明。**目的是为了别人和自己很容易看懂，一看就知道这段代码是做什么用的。

# 在Python中,#是一个注释符号

注释虽然出现在代码中，但机器在运行这段代码时并不会运行注释，而是跳过它。所以不必担心它会对我们的使用造成影响，同时也因为这个特性，我们可以利用它来屏蔽掉一些我们不想运行的内容。

```
import vapoursynth as vs
core = vs.get_core(thread=2)
```

这段代码的意思是，导入vapoursynth组件并将vs作为别名，并调用了get\_core这个方法，还将thread=2这个参数传递至get\_core，最后将结果赋值给core。

## 压制脚本调整

打开\_Link下**vsedit**，粘贴以下代码

```

#导入相关函数
import vapoursynth as vs
import kagefunc as kgf
import fvsfunc as fvf
import havsfunc as haf
import vsTAAmbk as taa
import mvsfunc as mvf
import muvsfunc as muf
import nnedi3_resample as nnrs
import nnedi3_rpow2 as nnrp

##性能配置
#获取CPU线程数(软件会自动使用所有线程)
core = vs.get_core()
#请依据自己实际内存调整(单位MB)
core.max_cache_size = 27384

##填写片源路径
source = r"J:\frdsnb.mkv"
#加载片源, 输入即为16bit色深
src = core.lsmas.LWLibavSource(source, format="yuv420p16")
#另一种加载片源方法, 并调整为16bit色深
#src = core.ffms2.Source(source).fvf.Depth(src, 16)

#切边
src = core.std.Crop(src, left=0, right=0, top=20, bottom=20) #数值必须为偶数

#脏边处理相关
#src = core.fb.FillBorders(src, 0, 1, 0, 0, mode="fillmargins")
#src = core.edgefixer.Continuity(src, left=4, right=4, top=0, bottom=0)
#src = haf.FixRowBrightnessProtect2(src, 1, +34)

#转为10bit再输出, x265一般压制为10bit, x264为8bit
src = fvf.Depth(src, 10)

#去交错(正常不需要使用, 多用于1080i及DVD)
#src = haf.QTGMC(src, Preset="slow", TFF=True)

#去锯齿
#src = taa.TAAmbk(src, aatype=-3, preaa=-1, strength=0, mtype=2, opencl=True)

##压制抽取 并与源对比
select = core.std.SelectEvery(src[8000:-8000], cycle=6000, offsets=range(80))
src = core.std.AssumeFPS(select, fpsnum=src.fps.numerator,
fpsden=src.fps.denominator)
#以上是抽取压制 正式压制时请删除或注释

src.set_output()

```

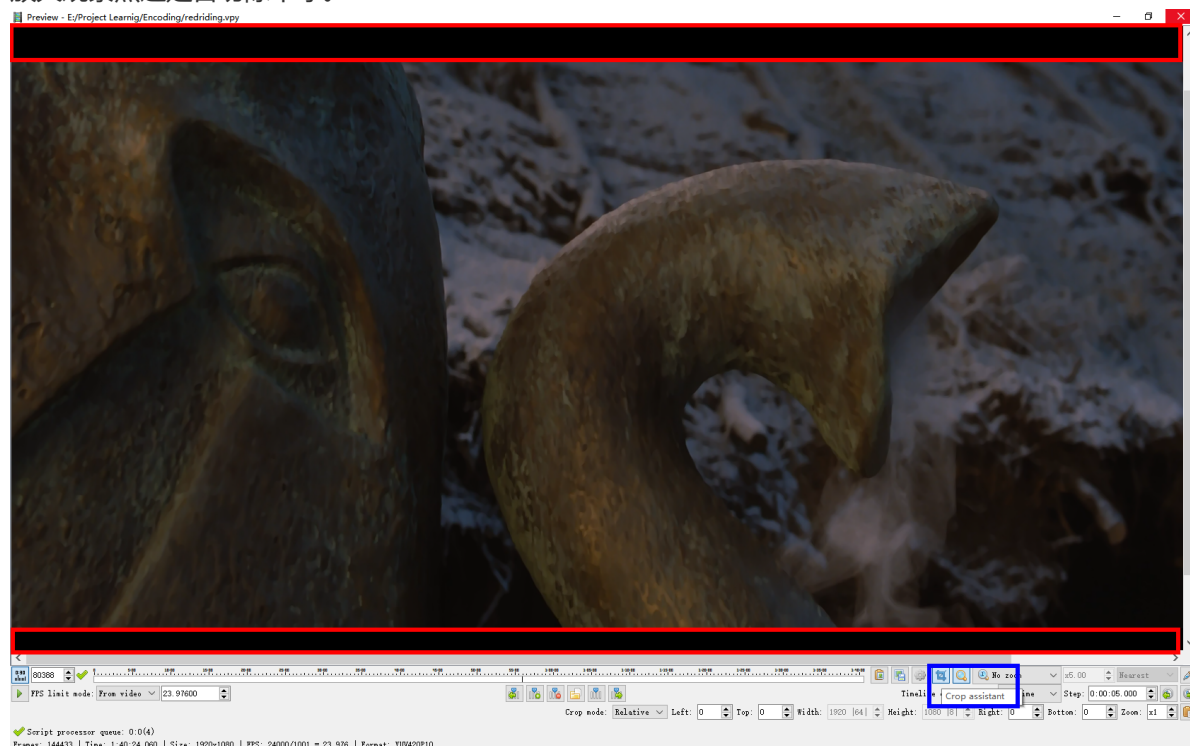
新手上路, 建议只修改**内存调整**、**加载片源**、**切边**这三部分代码。但不排除还需要添加其他代码进一步调整。

内存调整: 内存尽可能地给大, 例如分配6G, 即改为6148 (6\*1024)

加载片源: 将路径替换为自己的片源路径



切边：切除片源中带有的黑边(如图红框处)，点击右下角可以将右下角的 No zoom 改为 Fixed Ratio，放大观察黑边是否切除干净。



上图蓝框中为VSedit的切边助手，点击图标激活Crop Mode。在下方更改Left、Top、Right与Bottom数值，而后将参数回填至 core.std.Crop 函数中，或使用右侧的 Paste crop snippet into script，并修改脚本（需要一定代码基础）。

```
#Crop方法中，参数必须为偶数，黑边通常只出现在上下，即修改top与bottom参数
src = core.std.Crop(src, left=0, right=0, top=20, bottom=20)
#若遇到存在1像素这样的奇数边无法去除，可以补充一行脏边再切边。假设在底部有1行像素需要处理
src = core.fb.FillBorders(src, left=0, right=0, top=0, bottom=1,
mode="fillmargins")
src = core.std.Crop(src, left=0, right=0, top=0, bottom=2)
#若遇到存在1像素这样的奇数边无法很好的修复
src = core.resize.Spline36(src, 1920, 1038 src_left=0 , src_top=2 ,
src_width=1920 , src_height=1038)
```

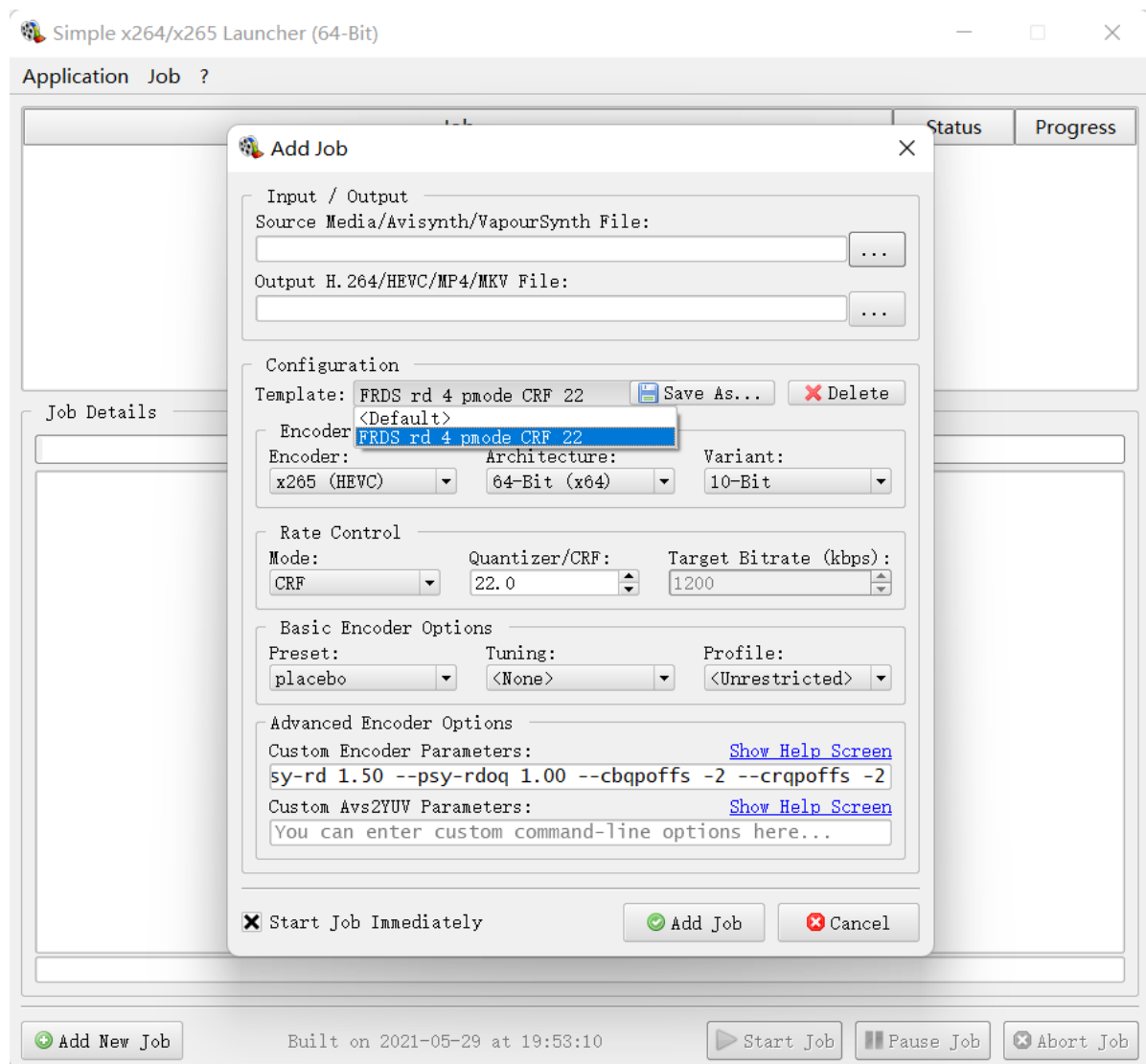
在编辑器内保存代码（本文举例为 encode.vpy），并且选择 Script-Preview 进行预览。第一次预览时需要创建 .ffindex/.lwi 索引文件，界面将会卡死无响应属于正常现象，可能较慢。

## 压制参数配置

打开\_Lnk下的x264\_launcher\_portable，选择 Application-Create New Job。

Source选择刚才保存的 encode.vpy，Output会自动输出 hevc 至与脚本相同的路径下，推荐使用预设模板 FRDS rd 4 pmode crf 21。





因提供的代码中使用了**压制抽取**，这部分将会抽取全片的其中一部分进行压制，通过生成的 .hevc 可以大致判断全片的码率及体积。通过调整crf数值，在17-23以0.5步进（如果机器性能较差可以以1为步进，找到差距较小的两个值(例如18与19)再取其中间值(例如18.5))进行测试，观察与原片在细节（纹理、边缘等）上是否有明显丢失，找到一个质量与体积的最佳平衡。一般推荐码率不超过源码率(原盘为输入源时的)50~75%，过高的码率将会使HEVC编码优势丧失。

这里引用进阶教程的测试流程，这里涉及到更多参数的调整，视自身情况操作。

通常，需要首先测试码率。只需对两个不同的CRF进行编码，并将它们与源进行比较，就可以找到接近于源质量的最高CRF值。现在，四舍五入这个值——最好是向下，并切换到2-pass。对于标准测试，测试qcomp(间隔0.05)，aq-mode，aq-strength(间隔0.05)，merange (32, 48, 64)，psyrd(间隔0.05)，ipratio/pbratio(间隔0.05并保持两者距离为0.10)，然后deblock(间隔为1)。如果你觉得mbtree有帮助（比如你在编码动画），启用mbtree重做这个过程。你可能不想过多地更改顺序，但确实可以这样做。

对于x265，顺序应该是qcomp、aq-mode、aq-strength、psy-rd、psy-rdoq、ipratio和pbratio，然后是deblock。

如果你想要更多一些效益，可以围绕着你为每个参数决定的值，用更小的间隔重新做测试。建议在对每个参数都进行了测试之后再行此操作，因为它们之间都存在轻微的相互影响。

一旦你完成了2-pass下的参数测试，切换回CRF并重复寻找最高视觉无损的CRF值的过程。

建议在测试后生成一组对比图进行对比。

制作对比图需要使用到一些代码，打开VapourSynth Editor，粘贴以下代码

```

import vapoursynth as vs
import kagefunc as kgf
import fvsfunc as fvf
import havsfunc as haf
import vsTAAmbk as taa
import mvsfunc as mvf
import muvsfunc as muf
import nnedi3_resample as nnrs
import nnedi3_rpow2 as nnrp
core = vs.get_core()

# 生成帧信息，并打入标签
def FrameInfo(clip, title,
               style="sans-serif,20,&H00FFFFFF,&H000000FF,&H00000000,&H00000000,0,0,0,0,100,100,0,0,1,2,0,7,10,10,10,1"):
    import functools
    def FrameProps(n, clip):
        clip = core.sub.Subtitle(clip, "Frame " + str(n) + " of " + str(
            clip.num_frames) + "\nPicture type: " +
            clip.get_frame(n).props._PictType.decode(), style=style)
        return clip

    clip = core.std.FrameEval(clip, functools.partial(FrameProps, clip=clip))
    clip = core.sub.Subtitle(clip, ['\n\n\n' + title], style=style)
    return clip

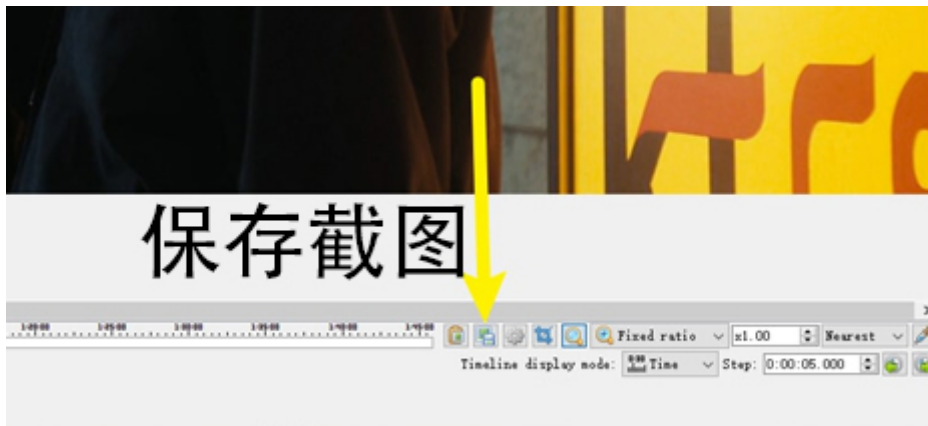
# VS editor输出显示bug，对于压制后的文件需要进行，进行处理。
def outfix(clip):
    encode = core.std.SetFrameProp(clip,prop="_Matrix",delete=True)
    encode = core.std.SetFrameProp(encode,prop="_Transfer",delete=True)
    encode = core.std.SetFrameProp(encode,prop="_Primaries",delete=True)
    return encode

#-----source文件-----#
video=core.lsmas.LWLibavSource(source=r"J:\frdsnb.mkv") # 载入源
#...同步压制脚本处理视频部分代码...#
video=core.std.Crop(video,64,64, 0, 0)
#...同步结束...#
video=fvf.Depth(video, 10)
video=FrameInfo(video,"source")# 标记信息
#-----encode文件-----#
encode=ccore.lsmas.LWLibavSource(source=r"J:\encode.mkv") # 载入编码后视频
encode=FrameInfo(encode,"encode") # 标记信息
encode=outfix(encode)# 输入修复

#-----输出交错预览-----#
out = core.std.Interleave([video,encode]) # 交叉帧
out.set_output()

```

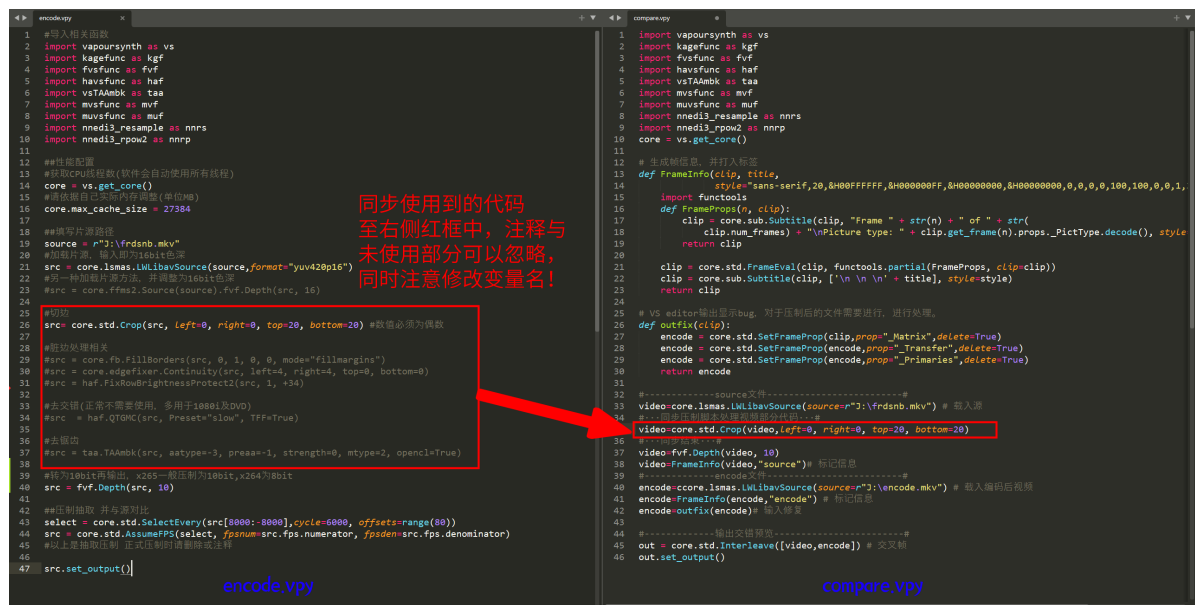
在编辑器内进行相应修改并保存代码（本文举例为 `compare.vpy`），并且选择 `Script - Preview` 进行预览。生成交错后，点击图中位置或右键影片保存截图，保存同一帧且 source 为 **P/B** 帧时 encode 为 **B** 帧的两张截图(按方向键←→切换)，推荐截取5帧共计10张图。



PS: 保存截图时可能出现同一数值帧两个源画面相差一帧的情况, 属于正常现象, 可以尝试调整压制源(video)为与encode源相同的输入函数, 若依然无法调整可忽略。

PPS: 很多人初次使用脚本时可能遇到 the 'clip' dimensions doesn't match... 的报错, 原因是对比图脚本中没有对source做相应的同步修改, 导致“画幅”不同无法生成。

例如压制时进行了切边操作, 相应地在对比脚本中也需要对source文件进行切边操作。



## 正式压制

确定脚本处理以及压制参数后即可开始正式压制。但正式压制前, 还需要修改之前使用的 `encode.vpy`, 将代码中**压制抽取**部分进行删除或注释。

Add Job 后, 就是漫长的等待, 并且电脑不能关机或睡眠。。。无聊时可以打开任务管理器看看满载的 CPU

如果出现跑不满的情况, 不必怀疑, 是你的CPU太强了! 需要多个任务同时方可吃满所有性能。

如需观看压制后的视频流(.hevc), 请使用VS载入并预览或者封装为mkv后播放, 播放器无法直接播放 .hevc。

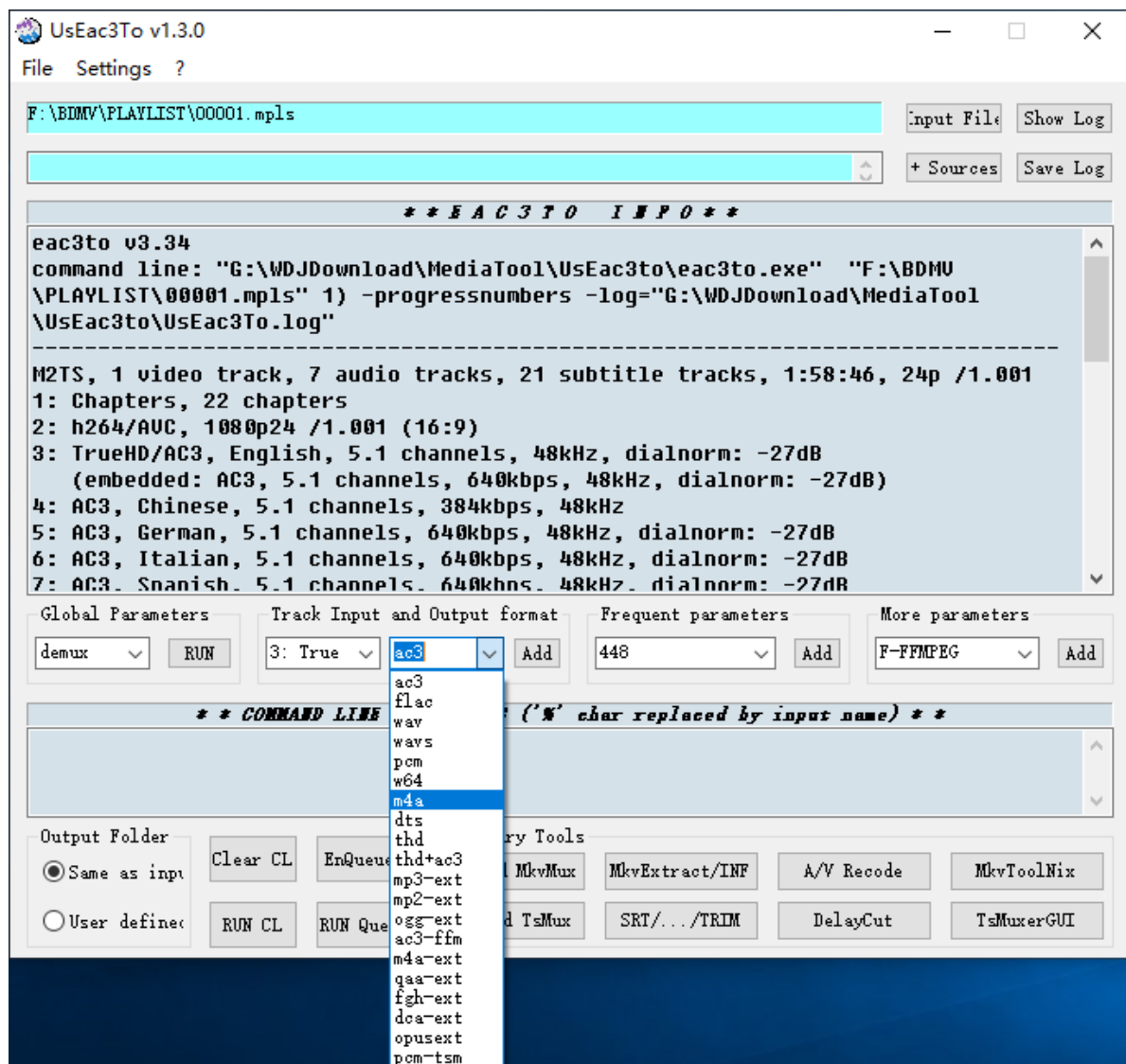
# 音视频字幕混流

## 音频处理

尽管源中的音轨形式多种多样，但主要使用 UsEac3To 进行音轨的提取，若源音轨符合要求可以直接输出。

打开\_Lnk下的**UsEac3To**，点击**Settings**，检查辅助工具的路径是否正确。如果需要设置输出文件夹(Output Folder)，其路径不要带有中文/特殊符号。

点击**Input File**，选择之前确定的 00001.mpls，随后会有一个命令行黑框出现，等待其读取结束，出现如图信息。



在**Track Input and Output**处选择需要提取的内容序号及输出格式，点击**Add**，**Command Line**会出现刚才添加的任务，而后点击**RUN CL**，即可开始提取。

以下简单列举可能会遇到的音频情况：

- 影片产地母语
  - **(推荐)**输出为DD+(EAC3)，近乎无损格式，适用于音轨为5.1/7.1声道的TrueHD / DTS / DTS-MA  
制作流程请参考[使用Dolby Encoder Suite制作DDP音轨教程] 参数设置请参考[韩小王大佬提供的截图](#)
  - 输出为AC3，有损格式，适用于音轨为DTS/TrueHD或原本已是AC3

- 输出为FLAC，无损格式，适用于音轨为2.0及以下声道的 PCM 或者 DTS-HD MA
- 影片非产地语言
  - (可选)输出为DD+(EAC3)，适用于音轨为非产地语言但为原盘版本的音轨或国配音轨
  - 输出为AC3，适用于音轨为非产地语言但为原盘版本的音轨或国配音轨
- 解说/评论音轨
  - 输出为AC3，适用于音轨为TrueHD/DTS\*的解说/评论音轨
  - 输出为AAC，有损格式，输出时添加码率参数为128，兼容性优于opus
  - 输出为opus，有损格式，输出时添加码率参数为128
- 演唱会

#### 情况复杂以下仅供参考

- **(推荐)**输出为DDP5.1/FLAC 5.1/TrueHD 5.1，尽可能地在音质与体积下权衡
- FLAC/aac 5.1，适用于音轨为PCM2.0

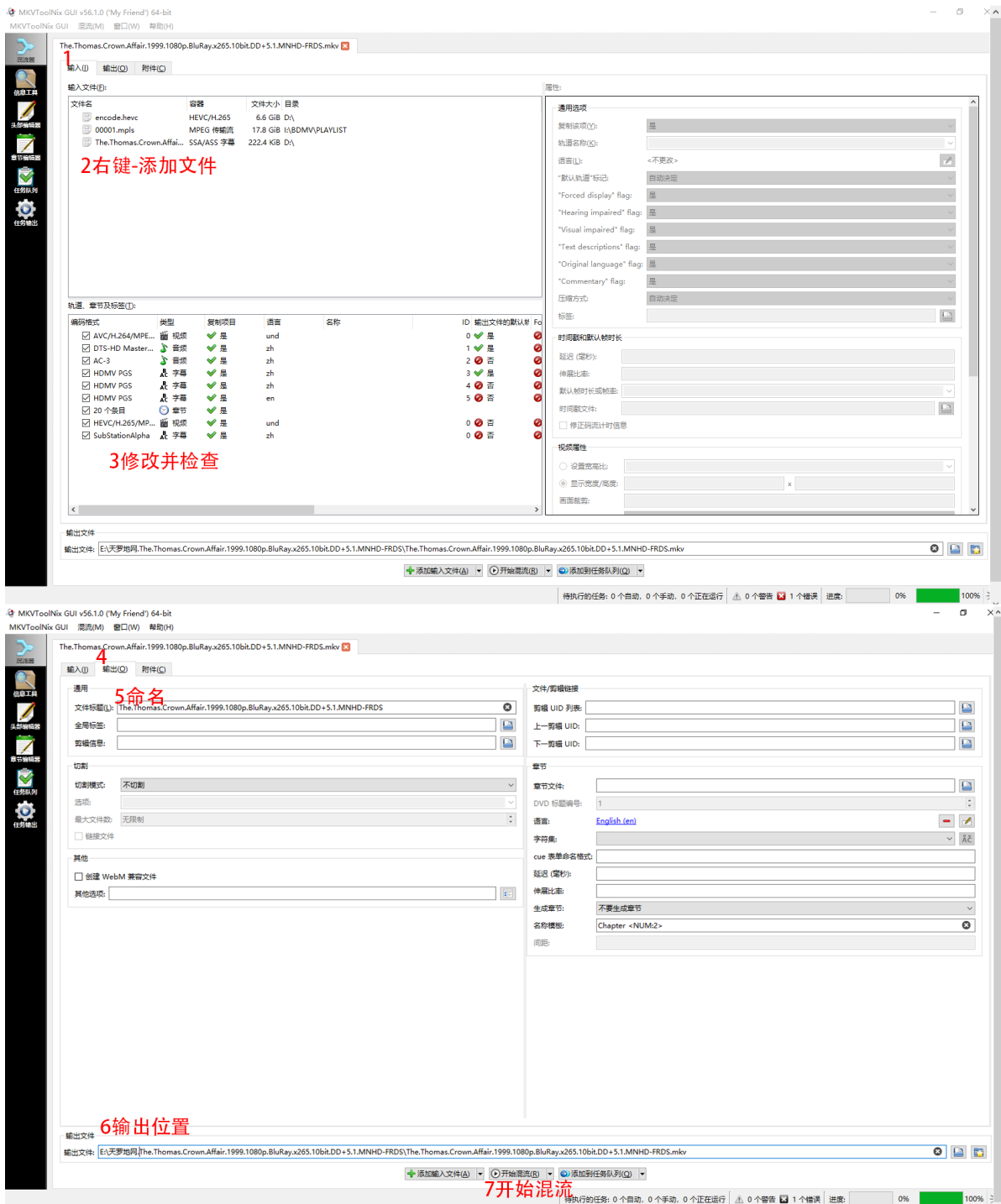
## 字幕获取

混流过程中不免遇到没有中文字幕的情况，虽然可以生肉发布，但还是建议添加中文字幕，推荐几个字幕网站

[SubHD](#) [字幕库](#) [伪射手](#)

找到字幕后请在播放器中载入字幕以核对其是否符合源。

## 混流



## • 添加文件

推荐将原盘作为输入源之一，方便直接导出其中的字幕轨以及章节信息。

## • 修改

源中可能预置了简体/繁体/听障友好字幕或普通话/粤语/导评音轨，都需要进行相应标注。

### ◦ 简体/繁体字幕

选中字幕，将简体中文字幕名称修改为CHS，繁体中文字幕名称修改为CHT。简中字幕名称建议修改为CHS&ENG，以此类推。

轨道、章节及标签(D):

编码格式	类型	复制项目	语言	名称	ID	输出文件的默认转
<input checked="" type="checkbox"/> HEVC/H.265/MPEG-H	视频	是	und		0	是
<input checked="" type="checkbox"/> AC-3	音频	是	und		0	是
<input checked="" type="checkbox"/> Opus	音频	是	und	Commentary	0	否
<input checked="" type="checkbox"/> SubRip/SRT	字幕	是	zh	CHS	0	是
<input checked="" type="checkbox"/> HDMV PGS	字幕	是	zh	CHT	0	否
<input checked="" type="checkbox"/> HDMV PGS	字幕	是	en		0	否

#### ◦ 听障友好字幕SDH

选中字幕，将右侧 "Hearing impaired" flag 改为是。

#### ◦ 导评音轨

选中导评，将右侧 "Commentary" flag 改为是，并且修改其名称为**Comment By XXX(XXX 为导评人)**。

属性	"Hearing impaired" flag:	否
1920x818 4	"Visual impaired" flag:	否
48000 Hz, 6	"Text descriptions" flag:	否
48000 Hz, 2	"Original language" flag:	否
48000 Hz, 2	"Commentary" flag:	是
48000 Hz, 2	压缩方式:	自动决定
1920x1080	标签:	
48000 Hz, 6		
48000 Hz, 2		
48000 Hz, 2		

#### ◦ 普通话/粤语音轨

选中音轨，将普通话音轨名称 **Mandarin**，粤语音轨名称标注为 **Cantonese**。

#### • 检查

确保视频、音轨、字幕和章节四者都有并保证顺序为 **视频-音轨-字幕-章节**，以及需要时各自名称/Flag有相应标注。

音轨推荐以 **原产地音轨-非原产地音轨-解说/评论音轨** 顺序排列

字幕推荐以 **简体-繁体-英语-其他语言** 顺序排列。

#### • 命名规范

输出时，请按以下格式进行命名。

- 初级命名推荐复制源名再按以下范例修改。<> 为必填，任何情况都必须有。

The.Thomas.Crown.Affair.1999.1080p.BluRay.x265.10bit.DD+5.1.MNHD-FRDS

即为 <影片名>.<年份>.<分辨率>.<来源>.<编码格式>.<位深>.<音频格式>.<压制组>

<音频格式>: 若为ac3格式则可以写为DDx.1(x为声道数)

- 进阶命名范例如下，适用于更加复杂的情况。() 为可选，视情况添加。

Zootopia.2014.GBR.Extended.Cut.1080p.BluRay.HDR.x265.10bit.DDP.5.1.MNHD-FRDS

即为 <影片名>.<年份>.(原盘版本).(版本信息).<分辨率>.<来源>.(视频效果).<编码格式>.<位深>.<音频格式>.<压制组>



(版本信息): 在原片存在多种官方版本(例导演剪辑、加长版、CC版)时建议填写。

(视频效果): 在视频带有HDR/DOVI(杜比视界)效果时建议填写。

(原盘版本): 原盘标注在不同国家发行的版本(例GBR、GER)时建议填写。

命名时字母与符号的使用需遵守以下:

允许的字符:

abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789.-

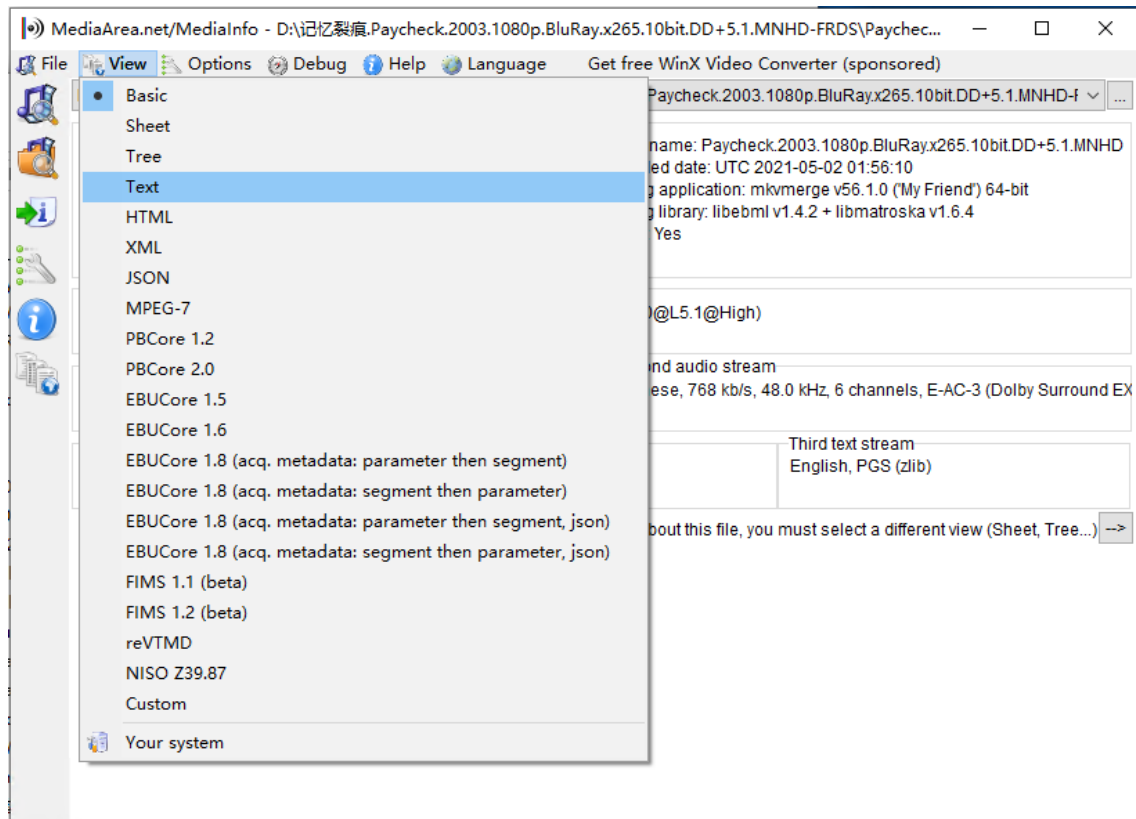
严格禁止的字符:

'!"\$%&\*()'

## 发布工作

### 1. 获取mediainfo信息

打开**MediaInfo**, 点击 **File - Open**, 选择你要读取的文件, 并点击 **View - Text**, 保存所有显示的内容。



或使用**PotPlayer**播放视频, 在画面中右键, **属性 - 文件信息**, 并点击左下角“复制到剪贴板”, 保存剪贴板内容。

### 2. 上传

上传图片需要使用到图床, 可以使用自己曾经使用过的图床或选择下方图床

[imgbox](#) (推荐使用代理上传) (推荐)[up.ccp.ovh](#)

上传完成后, 保存返回的直链, 即 <https://abc.dfe.com/hgk1mn.png>

### 3. 压制记录

在论坛回复[此帖](#)，记录压制相关内容，格式如下：

```
[quote]
压制片源：
<压制使用的片源名称>
[/quote]

[mediainfo]
<压制成品的mediainfo信息>
[/mediainfo]

[comparison=Source,FRDS]
<对比图脚本截取的图片>
https://xxxxxx.png
[/comparison]

[quote]
<压制时使用的参数，可以在simple x264压制结束生成的log中找到>
--crf 22 -D 10 --preset veryslow --high-tier --ctu 32 --rd 4 --subme 7 --ref
6 --merange=57 --me 3 --qg-size 8 --weightb --pmode --no-rect --no-amp --
rskip 0 --tu-intra-depth 4 --tu-inter-depth 4 --range limited --no-open-gop
--no-sao --no-early-skip --min-keyint=1 --rc-lookahead 100 --cutree --
bframes 8 --vbv-bufsize 160000 --vbv-maxrate 160000 --colorprim bt709 --
transfer bt709 --colormatrix bt709 --deblock -3:-3 --ipratio 1.3 --pbratio
1.2 --qcomp 0.65 --aq-mode 1 --aq-strength 1 --psy-rd 1.50 --psy-rdoq 1.00 -
-cbqpoffs -2 --crqpoffs -2
[/quote]
```

### 4. 制作种子

如果你正在使用的客户端支持做种，例如qBittorrent、utorrent，则可以直接制作种子，具体流程请[百度](#)。

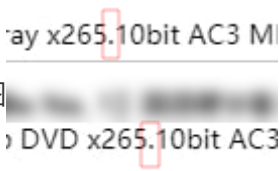
如果并不使用上述客户端，可以[dottorrent gui](#)、tranmission-create、[mktorrent](#)等工具制作，具体教程请百度。

使用到的tracker地址可以在 种子 - 发布 页面中看到。

### 5. 发布种子

在网站中点击 种子-发布，根据提示填写内容。实在不会填写，可以参考其他种子。

注意删除 x265与10bit之间的.，错误示范如图



PT-gen们: [Rhilip](#) / [BFDZ](#) / [IYUU](#) / [自建](#) / [电影信息查询脚本](#)

简介如下格式：

```
<影片信息（通过ptgen生成）>
[quote]
压制片源：<压制使用的片源名称>
感谢源发布者
[color=blue][size=4]
```

```
<源中的人为补充内容或压制时的其他修改例如字幕等
例：字幕来源于abc@YYeTs>
[/size][color]
[/quote]

[mediainfo]
<压制成品的mediainfo信息>
[/mediainfo]

[comparison=Source,FRDS]
<对比图脚本截取的图片>
https://xxxxxx.png
[/comparison]
```

点击**发布**！并在你的客户端上添加发布后的种子，校验后即可开始做种。

至此，你已经完成了一次压制工作！

## 结语

很高兴这个教程陪你完成了一次做鸭压制工作，但压制还有很多东西要学，在此推荐阅读以下[进阶教程](#)  
[压制高阶教程](#)

[VCB压制组出品教程](#) / [Encode Mystery](#)

[HDR影片参数计算方法1](#) / [HDR影片参数计算方法2](#) / [HDR影片参数计算方法3](#)

[脏线修复的各种方法](#)

[How to User eac3to](#) / [eac3to使用教程](#)

还有这些工具网站

[VideoHelp](#) 工具下载

[VSDB](#) VS插件库

[电影蓝光信息查询](#)

别忘了感谢过程中帮助过你的人~

## 引用

- [FRDS官方压制基本教程](#)
- [FRDS小组守则V0.2](#)
- [VapourSynth documentation](#)
- [Oday命名中字符规范](#)

## 更新日志

2021.7.30 V1

第一版，并于出版后几个小时后修正了导评音轨及非产地音轨的处理

2021.8.18 V1.1

感谢 @enjoyppdd 指正，抽取代码有误。经测试，SelectEvery产生的结果必须由一个新变量保存，再对其进行调用，否则会导致压制出错。

补充转码为FLAC的情况以及混流时音频字幕的顺序。

2021.9.15 V1.2

修正在VSedit中切边的操作及AC3音轨时的命名错误

修正在Oday命名中对国别的解释错误

添加1080p下通过resize切除奇数边的操作

压制抽取中画质的初级判断

若干细节补充和易化

2021.12.11 V1.3

感谢 @jrongbin 指正，对比图代码有误。经核查，缺少一条使video转换位置为10bit的代码导致片段不匹配，无法进行对比。

更改音轨输出判断

修正AC3音轨的命名

移动对比生成教程至测试压制

若干细节补充与纠正