

高品质 DVDRip 制作进阶教程

编撰 : DvDSharer

日期 : 2003-12-06

第一章 制作 DVDRip 的前期准备工作

第一节 工具软件介绍

- 1、Gordian Knot 0.28.5 - 集成工具包
 - 1) DVDecrypter - VOB 文件提取工具
 - 2) DVD2AVI - 音频分离工具
 - 3) AviSynth 2.5.2 - AVS 脚本工具 (包含 AVS 常用内置滤镜插件)
 - 4) Nandub - AVI 文件分割工具
 - 5) VobSub2.23 - 字幕工具
 - 6) BeSweet - 音频格式转化工具
- 2、AVI-Mux_GUI 1.15.3 - 视频+音频 (单/多音轨) 合成工具
- 3、XviD 24062003 - XviD 编码器
- 4、VirtualDubMoD 1.5.4.1 - 视频压缩工具
- 5、AviSynth 常用内置、外挂滤镜插件
 - 1) YV12 编码类
 - mpeg2dec3.dll
 - 2) IVTC/Dinterlace 处理类
 - decomb.dll
 - TomsMoComp.dll
 - 3) 降噪/平滑类
 - undot.dll
 - Convolution3dYV12.dll
 - MipSmooth.dll
 - deen.dll
 - edeen.dll
 - 4) 图像锐化类
 - MSharpen.dll
 - 5) 变形类
 - SimpleResize.dll
- 6、MPEG Custom Matrix - 自定义量化矩阵模板文件
 - 1) hvs-best-matrix
 - 2) hvs-better-matrix
 - 3) hvs-good-matrix
- 7、XviD Stats Viewer 1.65 - 1st pass 量化容量查看工具
- 8、GSpot 2.21 - AVI 信息查看工具
- 9、DRFAnalyzer 0.9.5 - AVI 品质分析工具

第二节 工具软件安装

- 1、运行 GordianKnot.RipPack.0.28.5.Setup.exe 自动安装指定的目录（建议安装在 D:\ 盘里）；
- 2、删除 D:\Gordian Knot\VirtualDubMod 下的所有文件，然后解压缩 VirtualDubMod_1_5_4_1_All_inclusive.zip，复制 VirtualDubMod_1_5_4_1_All_inclusive 文件夹内所有文件到 D:\Gordian Knot\VirtualDubMod 下；
- 3、运行 XviD-24062003-1 安装 XviD 编码器；
- 4、XviD Stats Viewer、GSpot、DRFAnalyzer、AVI-Mux_GUI 均是解压后直接运行的软件；
- 5、AVS 常用外挂滤镜插件解压后，复制 dll 文件到 D:\Gordian Knot 下即可。

第三节 软硬件环境

- 1、推荐频率在 G 级以上的 Intel 或 AMD CPU；
- 2、256MB 以上内存，DVD-ROM，10G 左右的自由硬盘空间；
- 3、建议使用 Win2000 专业版作为操作系统。

第二章 制作 DVDRip

第一节 提取 DVD 源文件

通常 DVD 光盘里的源文件是以 vob 文件的形式存放的，所以我们需要提取复制到硬盘上处理：在 Gordian Knot（以下简称：GK）中开启 DVDDecrypter，选择保存目录地址然后点击图标就自动解密将 DVD 的源文件复制到硬盘里了。工作完之后你会在目录下看到 DVD 的源文件。



图 1

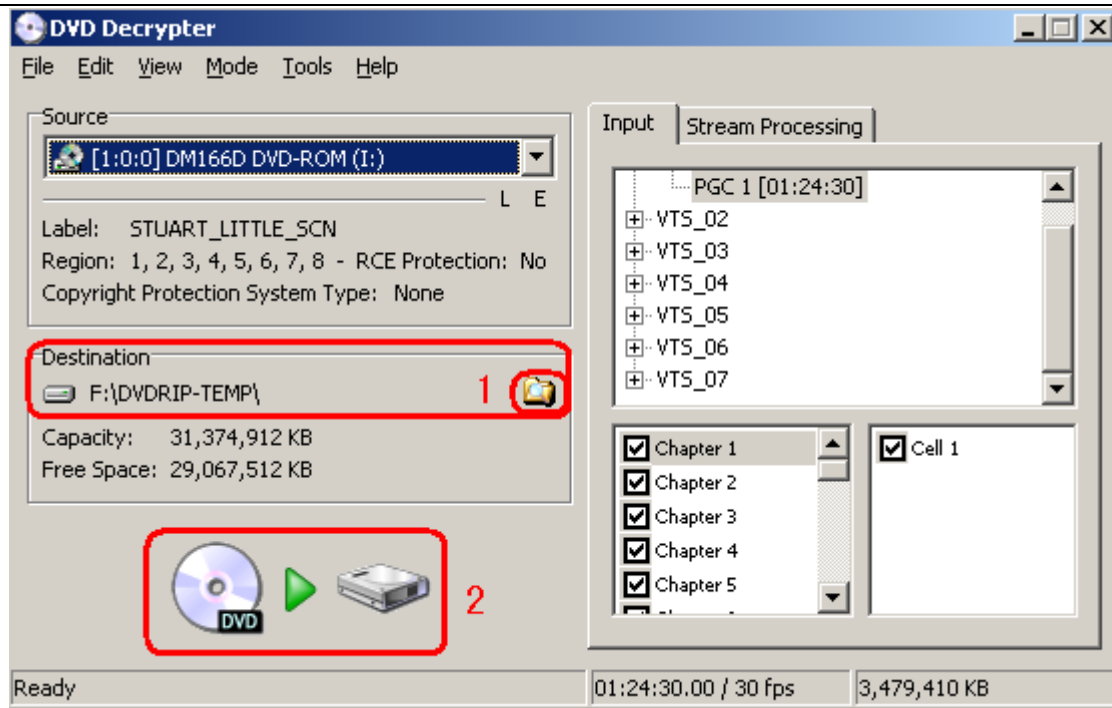


图 2

提取 IFO、VOB 文件后生成的 txt 文件，记载了所有 DVD 的相关信息可用来参考：

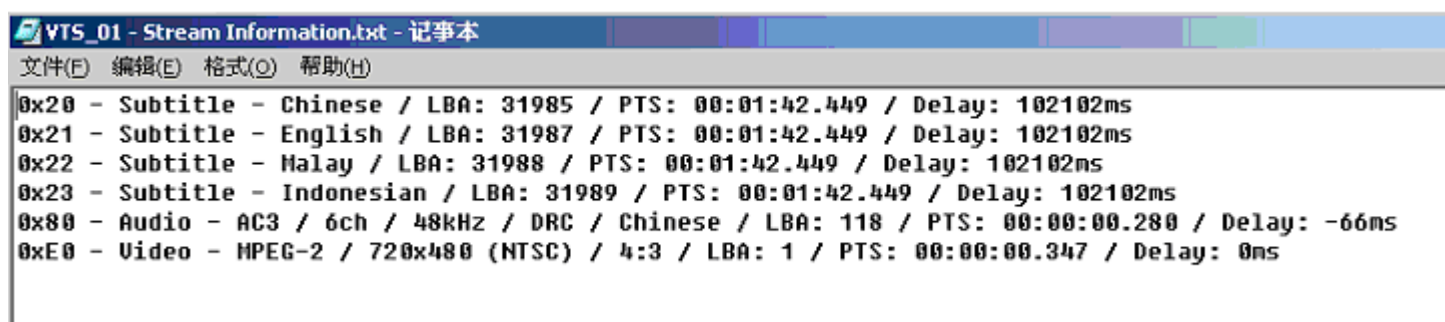


图 3

第二节 分离 VOB 源文件中的音频文件

1、我们在回到 GK 界面上并选取 DVD2AVI，

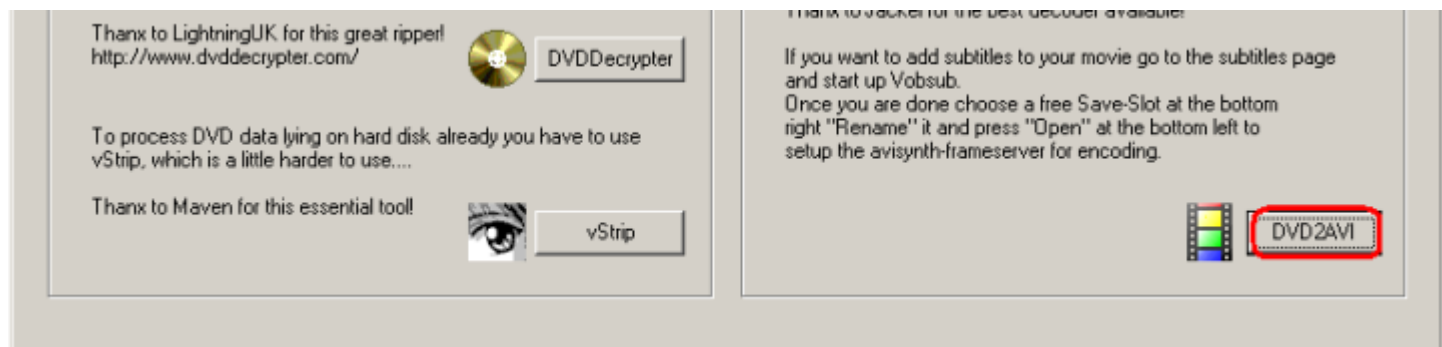


图 4

在 File → Open... 选择 VTS_01_1.VOB ,

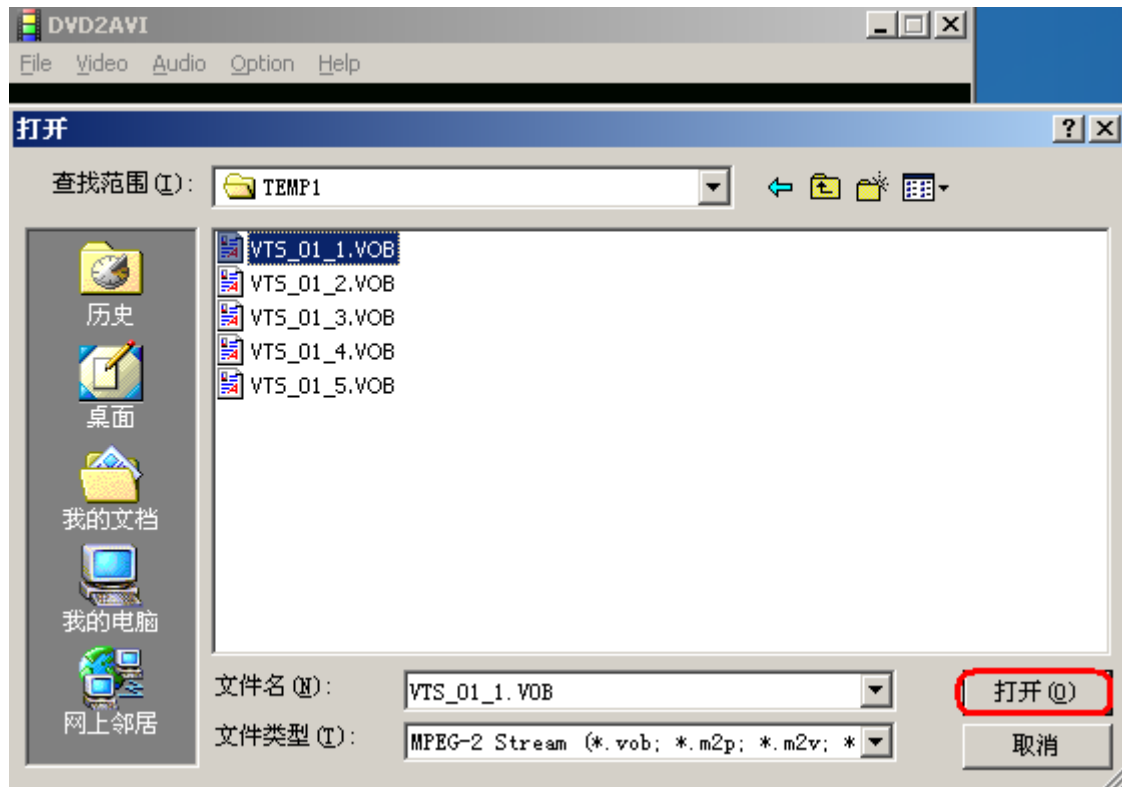


图 5

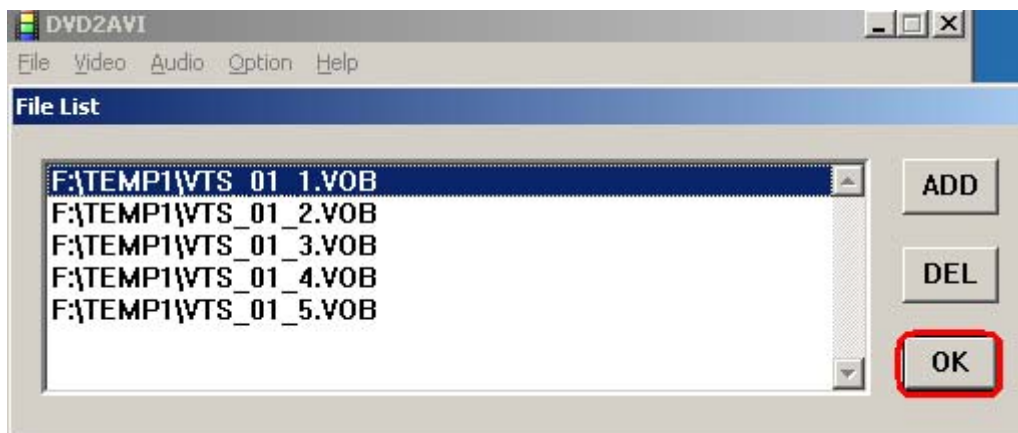


图 6

2、VD2AVI 相关设置原则

Video 设置：

- 1) iDCT Algorithm 设定为 IEEE-1180 Reference , 速度方面：32-bit SSE MMX > 64-bit Floating Point > IEEE-1180 Reference ; 质量方面：IEEE-1180 Reference > 64-bit Floating Point > 32-bit SSE MMX , 但由于 IEEE-1180 Reference 在某些机器上播放时会有算法方面的错误，所以在这里选择“64-bit Floating Point”
- 2) Field Operation 默认选取 None (可根据片源情况变化)
- 3) Color Space 里头选择 YUV 4 : 2 : 2
- 4) YUV --> RGB 则选择 PC Scale

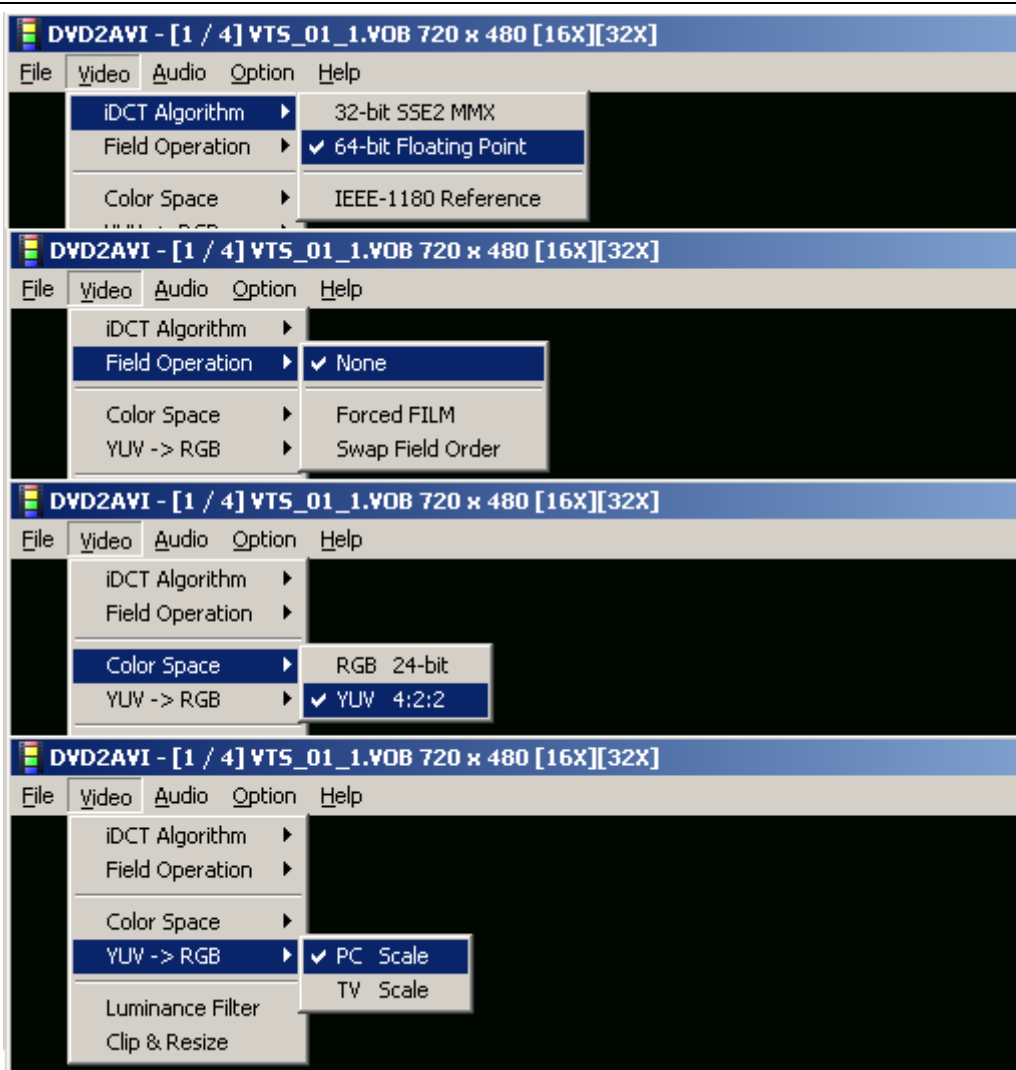


图 7

Audio 设置：

- 1) track Number → Track 1
- 2) Output Method → Demux All Tracks (AC3,MPA,DTS), 分离出 VOB 包含的全部音频文件
- 3) Dolby Digital Decode → Dolby Surround Downmix

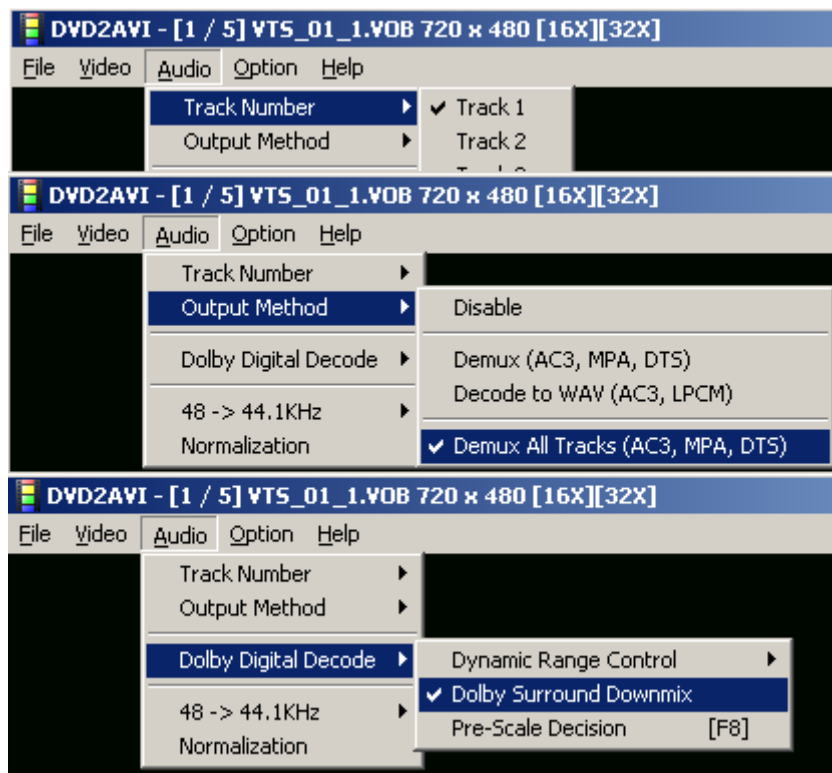


图 8

3、检测 VOB 的 Video Type 和 Frame Type，确定影片的 IVTC 或 Deinterlaced 处理方式：

按 F5 进行预览和检测一遍或大概跑一段（按 ESC 为中止预览）

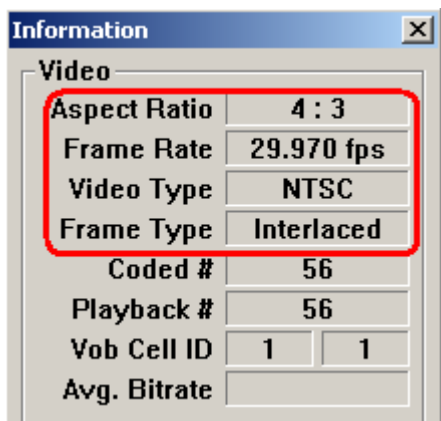


图 9

1) NTSC Interlaced 型（常见类型）：

Frame Type 为 Interlaced，说明包含有交错场，是隔行显示的，需要利用 IVTC 做去场处理（后续步骤有详述），以还原成影片 23.976fps（24fps）的原始帧率，Field Operation 默认选取 None；

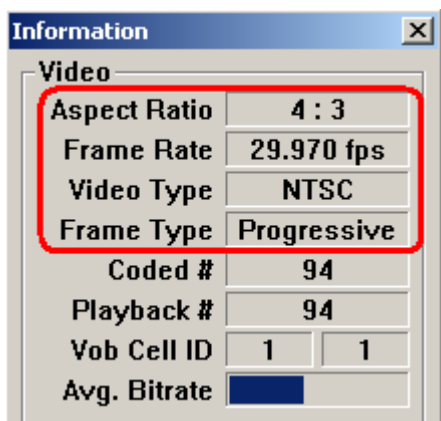


图 10

2) NTSC Progressive 型（少见类型）：

Frame Type 为 Progressive，说明是逐行显示的，不需要做任何去场处理，必须要保持原始的 29.970fps 帧率，Field Operation 默认选取 None；

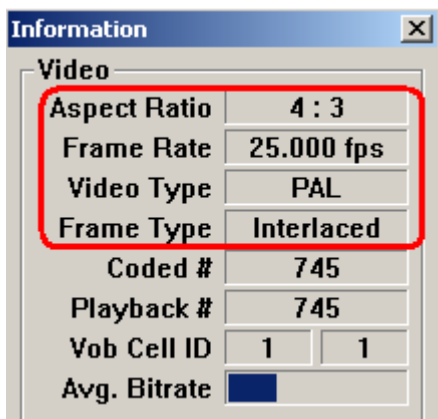


图 11

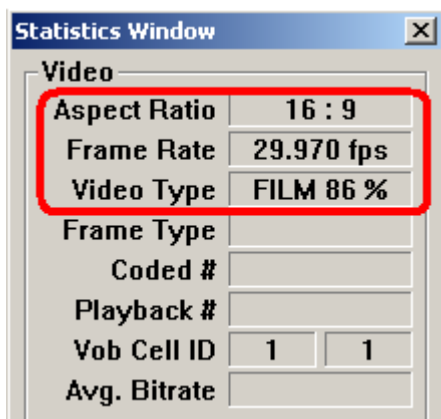


图 12

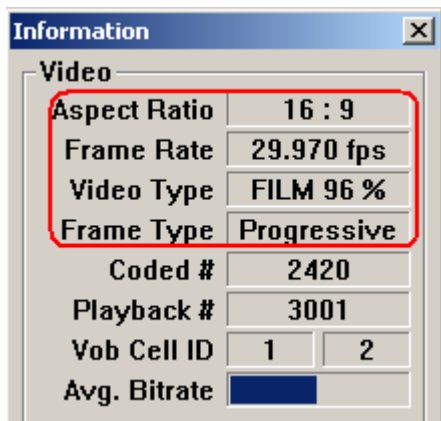


图 13

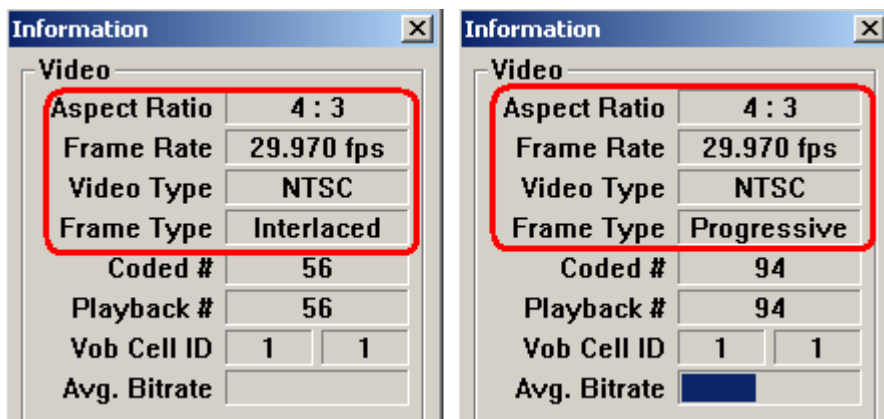


图 14

3) PAL Progressive 型 (常见类型):

虽然 Frame Type 提示是 Interlaced, 但对于 PAL 的片源仍需要再次确认。方法如下:

- 预览时注意观察运动画面是否水平黑色扫描线, 如果没有, 则实为 Progressive 型;
- 如果有, 则中止预览, 选取 Field Operation → Swap Field Order 变换场序后, 再次进行预览检测, 仔细观察是否仍有水平黑色扫描线。如果没有则是逐行显示的, 不需要做任何处理; 如果有, 则要进行 Deinterlaced (后续步骤有详述)。Field Operation 始终默认选取 None。

4) NTSC FILM 型 (常见类型):

Video Type 为 FILM XX% 或 FILM, 如果 $XX\% \geq 95\%$ 或 Frame Type 为 Progressive, 说明 DVD 片源是以规则 RFF 旗标将 24fps 的 film 信息保存于 DVD 光盘中, 然而输出的时候, DVD Player 会根据 DVD 内的 RFF 旗标的信息, 将内部储存的 24fps Frame 组合成 30fps 输出, 以符合 NTSC TV 的规格。所以 DVD2AVI 的 Force Film 就是根据这两个旗标的信息, 反过来不做 24→30 的转换, 直接输出原本内部储存的 23.976fps (24fps)。即 Field Operation 选取 Film Force;

如果 $XX\% < 95\%$ 说明 DVD 片源的存储 RFF 旗标是不规则的, 也就是 24fps 的比例不高, 为 24/30 fps 混杂的 DVD, 强制 Film 输出的结果会得到错误的 Frame 组合, 所以在需要利用 IVTC 做更精确的去场处理 (后续步骤有详述), 此时 DVD2AVI 设置应先保持 29.970fps (30fps) 的原始帧率, 即 Field Operation 默认选取 None。

5) NTSC Interlaced / Progressive 混合型 (极少见类型):

Video Type 为 NTSC, Frame Type 为 Interlaced 和 Progressive 交替变换的, 说明片源是以 29.970fps 的逐行和交错两种帧型混合灌制的。第一, 有可能是盗版的时候没做 decss 或者直灌而是用模拟信号输出采集制作而导致的; 第二, 欧美电影中的确少见, 一般都是 film 的, 偶尔会在剪辑的地方出现 ntsc interlaced, 但在日本动画中极为常见。最近的 WHR 和 MZ 等大作也都这样。一般是由于分段处理造成的, 一部分普通的可能被做成了 film, 还有一部分比如有 3d 背景的, 或者做完后重新编辑插入字幕的, 就被做成 ntsc interlaced 了。但应该都可以做 ivtc 的, film 的基本是 RFF, 重复了第一帧, ntsc interlaced 的是 3:2pulldown 的, 所以 DVD2AVI 时仍应保持初始的 29.970fps, 后续全部做 IVTC 为 23.976fps (后续有详述), Field Operation 按照默认设置选取 None。

4、确定好 DVD2AVI 的各项设置后, 最后 Save Project..... 保存 d2v 工程文件, 工作完成之后你会看到在目标目录下, 分离了*.d2v 和*.ac3 的档案。

如: 1.d2v

1 AC3 T01 2_0ch 192Kbps DELAY -66ms.ac3

1 AC3 T02 3_2ch 448Kbps DELAY -66ms.ac3

第三节 分析视频容量和分辨率以及 AVS 脚本测试

1、利用 GK 计算最终 AVI 中的 Video 容量、裁边 (Crop) 分辨率 (Resize)

1) 打开 GK, 导入先前 DVD2AVI 生成的 d2v 文件,

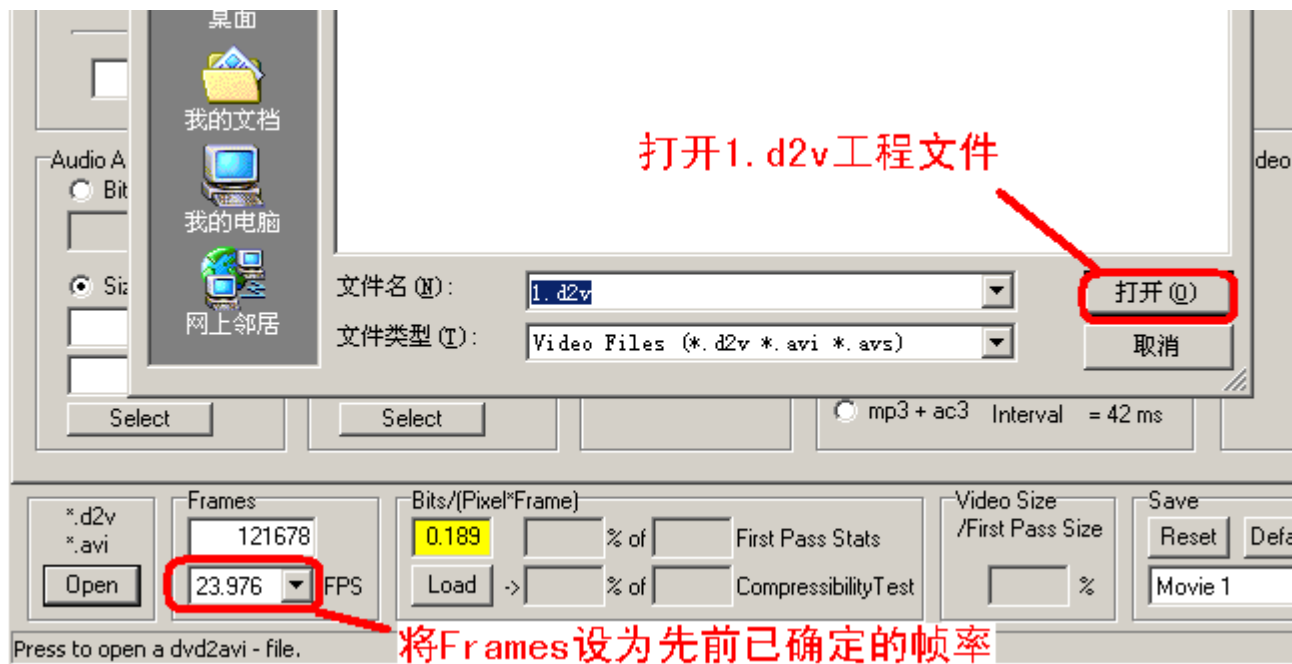


图 15

接着我们在回到 GK 的 Bitrate 界面上，我们先来计算码率并决定要做成多少 CD 数版本。

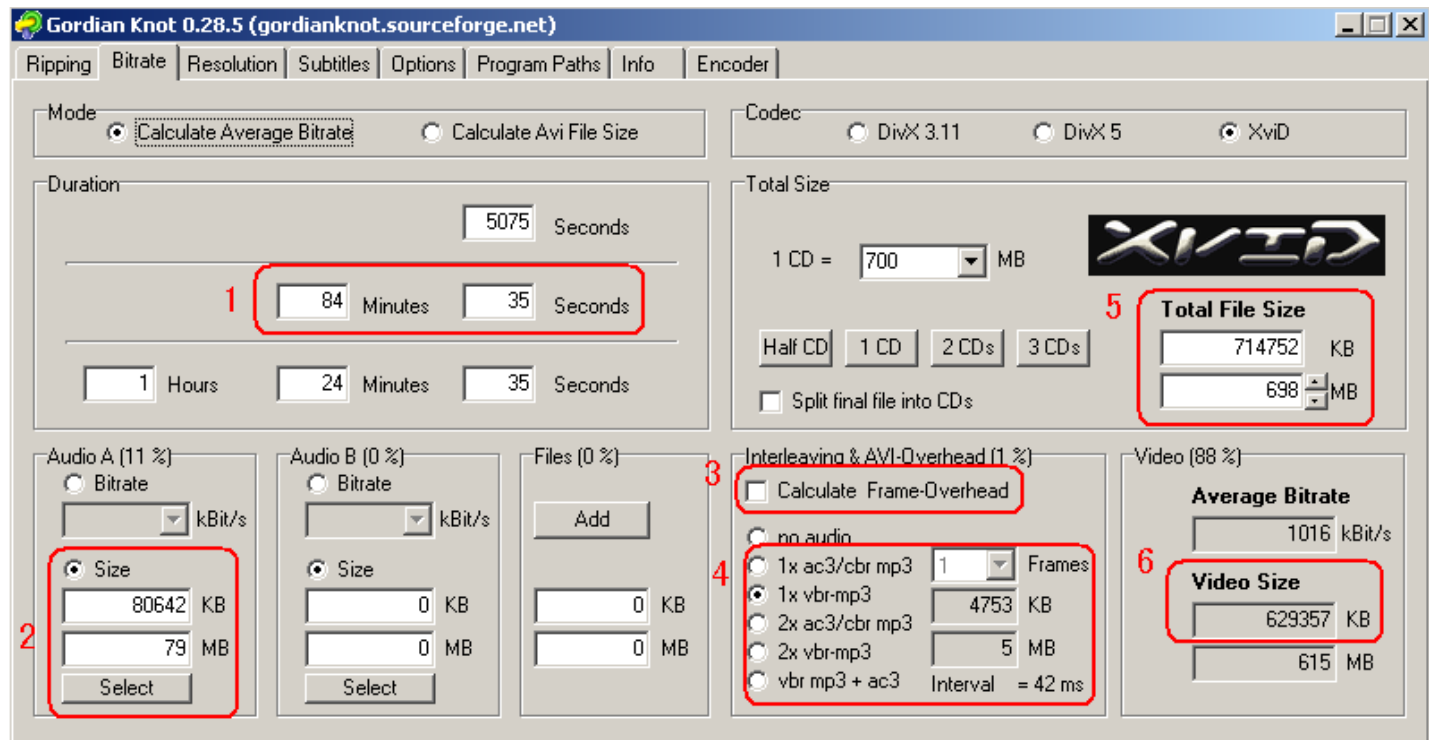


图 16

当我们开启*.d2v 的文件时，GK 就会自动的把整部片子的正确时间点填入了 → 选取 AC3(or MP3)音频文件 → Calculate Frame-Overhead 项不要选(可以增加些 Video 码率) → 选择 1x ac3 若你的 AC3 标示 192k/224k/256k 那你就选择 3 若是 384k/448k 就选择 2，若是 MP3 就选择 1x vbr-mp3 → 输入期望的 AVI 容量为 698MB(1CD)或 1394MB(2CD)或更多。。(这里注意不要设置满了，要考虑给字幕预留一部分空间，例如：1CD 不要设为 700MB，2CD 不要设为 1400MB) → 记录 Video 容量 (xxxxxKB)，以备后用。

备注：AC3 转 MP3 音频方法详见本章第五节内容。

2)接下来我们直接回到 Resolution 的界面上,到这里先暂停下,我们先来了解一些裁切(Corp)和变形(Resize)的理论常识:

首先说变形,所有的 DVD 都是把电影胶片按照两种变形方式来录制的,4:3 和 16:9,没有 2.35:1,2.35:1 实际是 16:9 再裁掉上下黑边而来。

DVD 都是 720X480(NTSC) 或 720X576(PAL) 的画面尺寸。我们假设所有的 AVI 最大宽度为 640。以 NTSC 制式 DVD 为例:

对于 DAR=4:3 的 DVD 正确步骤是:

- 1、左右共 Crop 16 个点,720X480->704X480
- 2、因为宽度是 PAR=1.1:1 取样,进行还原 704X480->Resize 成(704/1.1=640)X480
- 3、再 Resize 成需要的画面尺寸,宽度取 16 的倍数,高度 = 宽度/DAR,例如:640 X 480、624 X 464、608 X 448、592 X 432、576 X 432、544 X 408、512 X 384、等,但由于是满屏显示需要的像素点也是最多的,为了节省有限的视频码率,所以我们通常选择 512 X 384 或 480 X 352 两种分辨率。
- 4、裁掉多余的上下左右黑边。

对于 DAR=16:9 的 DVD 正确步骤是:

- 1、左右共 Crop 16 个点,720X480->704X480
- 2、因为宽度是 PAR=1.1:1 取样,进行还原 704X480->Resize 成(704/1.1=640)X480
- 3、再 Resize 成需要的画面尺寸,宽度取 16 的倍数,高度 = 宽度/DAR,例如:640 X 352、624 X 336、608 X 336、592 X 320、576 X 320、560 X 304、544 X 304 等。
- 4、裁掉多余的上下左右黑边。

对于 2.35:1 画面的 DVD 实际是按照 16:9 做的,只是裁的时候上下多一些,比如正常情况下 640X360 裁完了上下黑边后出来高度是 272。 $640:272=2.35:1$,这就是大家常说的 2.35:1 画面 DVD。

验证最后的画面尺寸,需要保持宽度和高度是 16 的倍数(TDX2002 已明确规定)。不是准确的倍数的话有两种办法,一是继续裁,一是补黑边。一般来说约定俗成的做法不喜欢有黑边,所以一般选继续裁。

实际在做裁切(Corp)和变形(Resize)时,由于种种原因怎么裁切都无法使得变形市镇率为 0%,因此对于肉眼是无法辨识的轻微变形失真(Asspect Error < 1%)的情况是允许的。

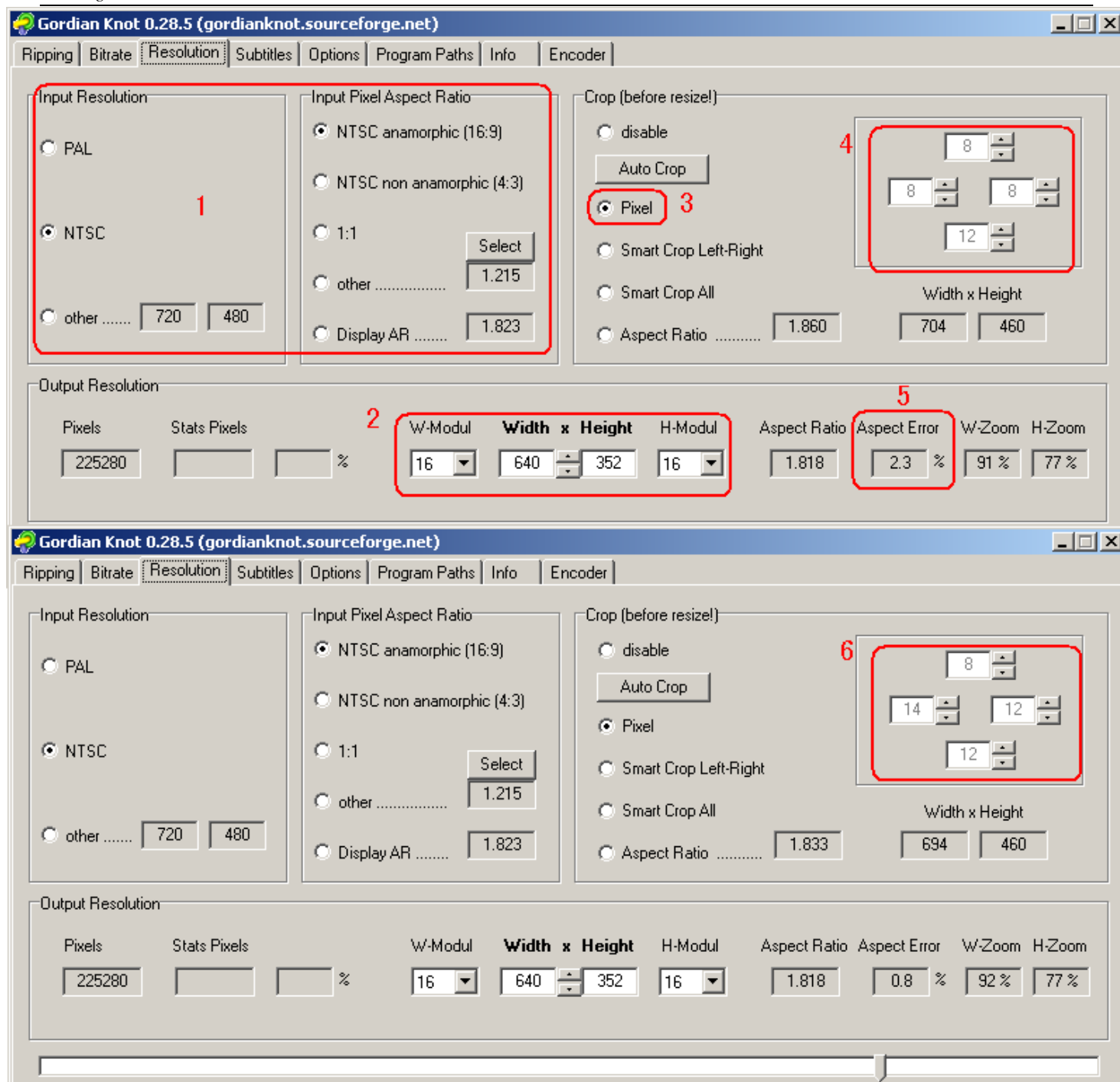


图 17

3) AVS 脚本文件测试：

前面的 Corp 和 Resize 初步设定完成后，即将开始进行一项非常有意义的工作：手工创建、编辑以及测试 AviSynth 执行脚本文件（以下简称 AVS 文件）

首先，用新建一个 TXT 文本文件，然后将文件后缀名更改为 AVS。用记事本打开 AVS 文件，进行编辑，以下我给出一个 AVS 文件基本格式实例，其中 # 号后面为注释讲解：

```
SetWorkingDir("d:\GORDIA~1\") # 设定 AVS 脚本调用 dll 插件文件的工作路径

LoadPlugin("mpeg2dec3.dll")      # 调用编码器插件

LoadPlugin("decomb.dll")         # 调用交错滤镜插件，包含了 IVTC 和 Deinterlace 两种滤镜

LoadPlugin("TomsMoComp.dll")    # 调用交错滤镜插件，对应 Deinterlace 滤镜中的 TomsMoComp 脚本命令

LoadPlugin("SimpleResize.dll")  # 调用变形滤镜插件，对应 SimpleResize 脚本命令

mpeg2source("F:\temp1\1.d2v")    # 源 d2v 文件的路径

trim(startframe,endframe)       # 截取 VOB 片断进行压缩。startframe 为起始帧，endframe 为结束帧。主要用于片断测试
                                # 压缩或更高级的分段处理压缩

Telecide(guide=1)               # IVTC 滤镜。多用于做 NTSC 制式 DVD 的 IVTC 处理

Decimate(Cycle=5)               # IVTC 滤镜。用于 NTSC 制式 DVD 删除重复帧处理，即 29.970fps --> 23.976fps

FieldDeinterlace()              # Deinterlace 滤镜。用于做 PAL 制式 DVD 的 Deinterlace 处理，不能同 IVTC 和 TomsMoComp
                                # 滤镜同时使用

FieldDeinterlace(blend=false)   # 同上

TomsMoComp(1,5,1)               # Deinterlace 滤镜。在处理 PAL 制式 DVD 时，一般不能同 FieldDeinterlace 和 Telecide 同时
                                # 使用

crop(14,8,-12,-12)              # 裁切命令。参数为（左、上、右、下），代表各边被裁切去的像素值，这组参数可以根据
                                # 图 17 中第 6 框中的示数来输入

LanczosResize(640,352)          # 变形滤镜。确定最终 AVI 的分辨率，常用于普通电影

SimpleResize(640,352)           # 变形滤镜。确定最终 AVI 的分辨率，常用于二维动画，不能和 LanczosResize 同时使用

SelectRangeEvery(xxx,14)        # 全片平均采样测试语句。其中 xxx 主要用于 COMPRESSIBILITY CHECK 测试，即
                                # (2pass/1pass)*100% 预测算，而正式压缩时不用。其中 xxx 为平均采样点的间隔值。例如当
                                # 进行全片 5% 预测，则 xxx=280，10%为 140，15%为 93
```

备注：若不想执行某条语句，只需在该条语句前标记 # 字符即可

在本章的第二节第3点中提到“根据检测 VOB 的 Video Type 和 Frame Type，确定影片的 IVTC 或 Deinterlaced 处理方式”，这里我们就可以编辑 AVS 脚本命令来添加需要的 VOB 处理滤镜，原则如下：

A、对于 NTSC Interlaced 型：需要做 IVTC 和去重复帧处理，则 AVS 脚本应包含 Telecide(guide=1) 和 Decimate(Cycle=5) 语句。

B、对于 NTSC Progressive 型：不做任何处理，则 AVS 脚本无需包含 IVTC 和 Deinterlace 滤镜。

C、对于 PAL Progressive 型：不做任何处理，则 AVS 脚本无需包含 Deinterlace 滤镜。

D、对于 PAL Interlaced 型：需要做 Deinterlace 处理，则 AVS 脚本应包含 FieldDeinterlace() 或 FieldDeinterlace(blend=false) 或 TomsMoComp(1,5,1) 语句。

E、对于 NTSC FILM 型：若为 纯 FILM 或 FILM 95% 时，则在 DVD2AVI 阶段就已经过 force film，这里就无需处理了；若 FILM < 95% 时，需要做 IVTC 处理，则 AVS 脚本应包含 Telecide(guide=1) 和 Decimate(Cycle=5) 语句。

F、对于 NTSC Interlaced / Progressive 混合型：全部做 IVTC 处理，则 AVS 脚本应包含 Telecide(guide=1,Gthresh=50,dthreshold=8,threshold=0,Post=true).Decimate(5) 语句。

然后，设定好 d2v 文件的路径，输入先前确定的 crop 和 resize 参数，加入全片 5% 的 SelectRangeEvery(280,14) 预测算语句（当然你也可以进行 10%、15% 的预测算，只要修改 xxx 值即可。总之采样的比率越大测算的结果就越精确，但同时所花费的时间也越长）。

```
SetWorkingDir("d:\GORDIA~1\")
LoadPlugin("mpeg2dec3.dll")
LoadPlugin("decomb.dll")
mpeg2source("F:\temp1\1.d2v")
telecide(guide=1)
Decimate(Cycle=5)
crop(14,8,-12,-12)
LanczosResize(640,352)
SelectRangeEvery(280,14)
```

保存 AVS 文件，打开 VirtualDubMod（以下简称 VDM）程序，导入 AVS 文件。

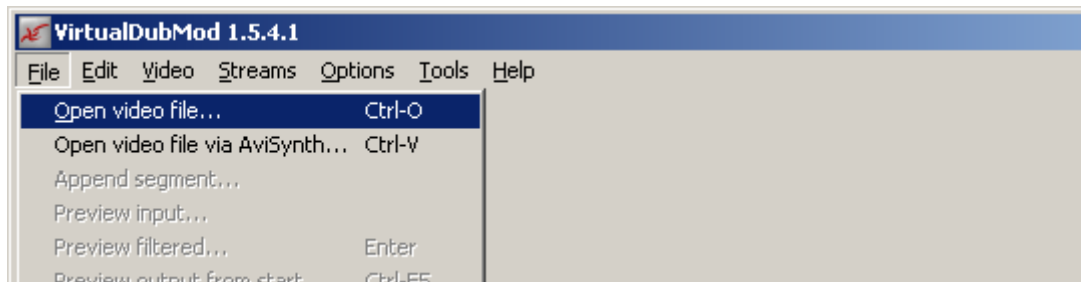


图 18

这里讲一下 VDM 的基于 Xvid 的 1st pass 和 2nd pass 压缩的相关原理：

1、Xvid 的 I-frame、P-frame、B-frame:

IF 就是关键帧，是一个帧组的第一个帧，帧组就是 GOP，Group of Picture。IF 保留了一个场景的所有信息。PF 是未来单向预测帧，只储存与之前一幅已解压画面的差值。BF 是双向预测帧，除了参考之前解压了的画面外，亦会参考后面一帧中的画面信息。IF 压缩比在 1:7，PF 是 1:20，BF 是 1:50。因为 BF 是双向预测帧，在前后的 PF 进行预测，所以比较慢，解码也慢，对播放的机器配置有一定的要求。

2、1st pass 和 2nd pass:

为什么要做 1pass 和 2pass？1pass 就是用 Quantizer 值 2 把 VOB 预先压一遍，但 Q 值 2 显然会文件体积超标（用 XviD Stats Viewer 可以看到），然后根据分析结果和目标大小来合理安排码流，低动态的稍微少分一点，大动态的多分一点，总之保证目标大小。现阶段 Xvid 的动态分配（即曲线压缩）做得不好，所以直接采用了一种 linear scale（线性压缩）来把 1pass 的码率曲线直接往下调。

1、关于 1st pass 设置：

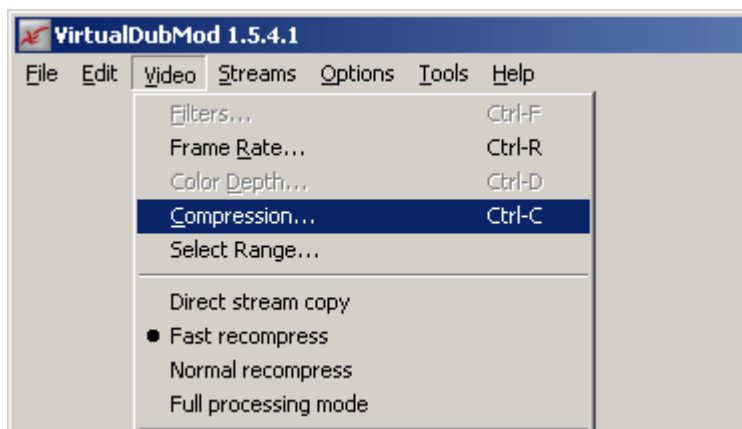


图 19

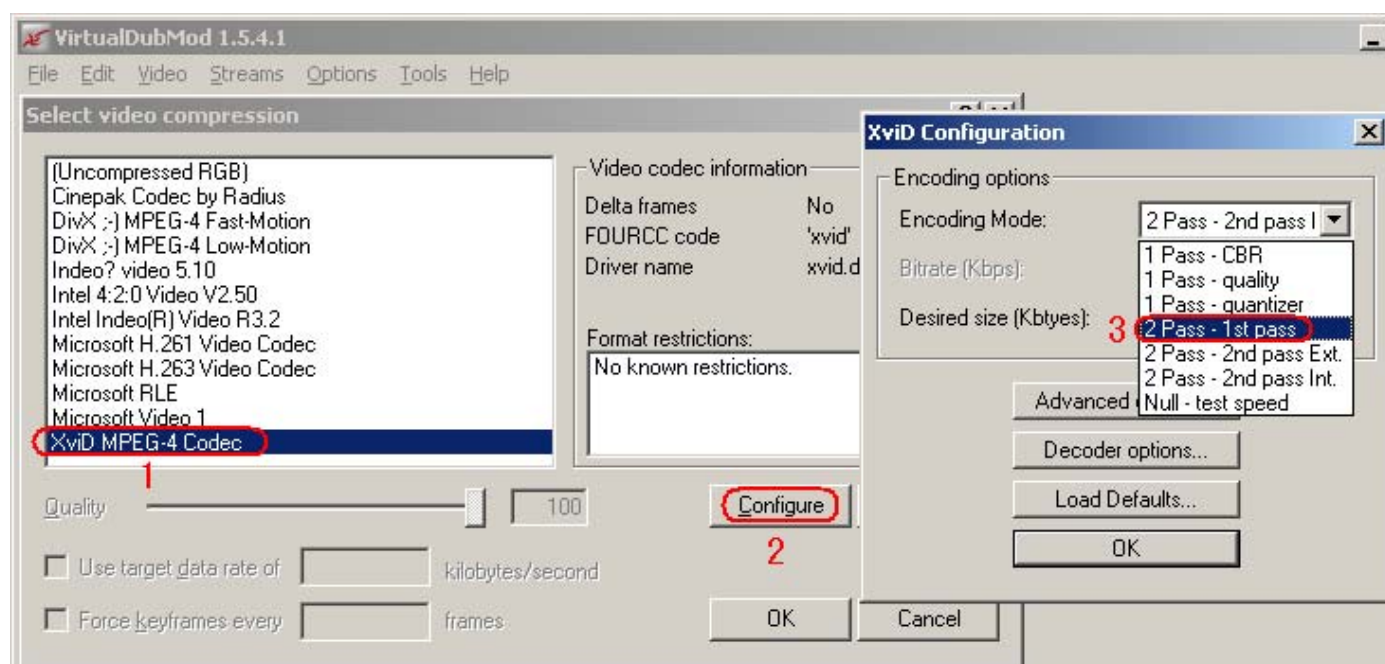


图 20

选择好“2Pass - 1st pass”后，点击 Advanced options...

Global settings:

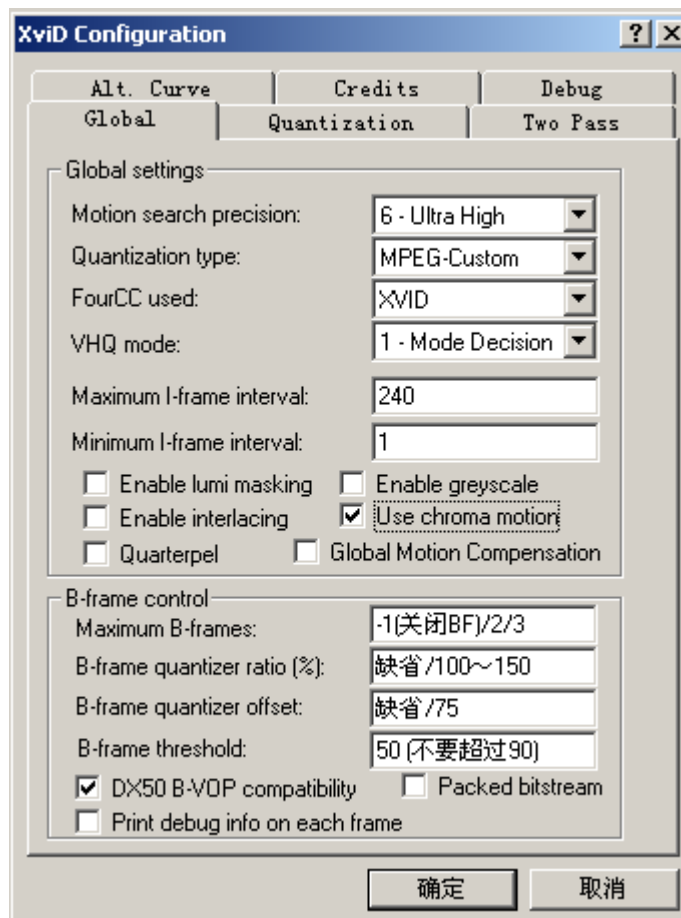


图 21

1) Motion search precision(运动检测精度):运动检测精度决定着给定码流下的画面质量。一般都选 6-Ultra High，本着出好作品的原则，5-Very High 就不要了，只快 10%。

2) Quantization type(量化方式):此处应该根据目标码流来决定。高码流（1100kbps 以上）建议使用 MPEG 量化方式，可以得到最佳结果。如果码流很低（700 ~ 900kbps），建议使用 H.263 量化方式。MPEG 方式比 H.263 画面细节质量好。但根据一些高手的大量试验和我的一些经验，用 hvs-matrix 系列量化模板可以兼顾细节和降噪的双重优点，MPEG、H.263、hvs-matrix 系列（hvs-best、hvs-better、hvs-good）量化相同 VOB 所产生的 1pass 文件容量排序：MPEG > hvs-best > H.263 > hvs-better > hvs-good

由此可见，MPEG 产生的 1pass 文件最大，保留细节也最多，但相对也最难压缩，消耗的视频码率也最多；hvs-best 和 better 都很不错量化产生的 1pass 文件和 H.263 差不多，确可以保留比 H.263 更多的画面细节；hvs-good 的产生的 1pass 最小，保留细节只略少于 H.263，但对于非常难压的 VOB 则是非常有效果的。

用法：若用 H.263 和 MPEG 方式可以直接选取；若用 MPEG Matrix 方式量化可以选择 MPEG-Custom，然后按照下图调用 Matrix 模板文件即可。

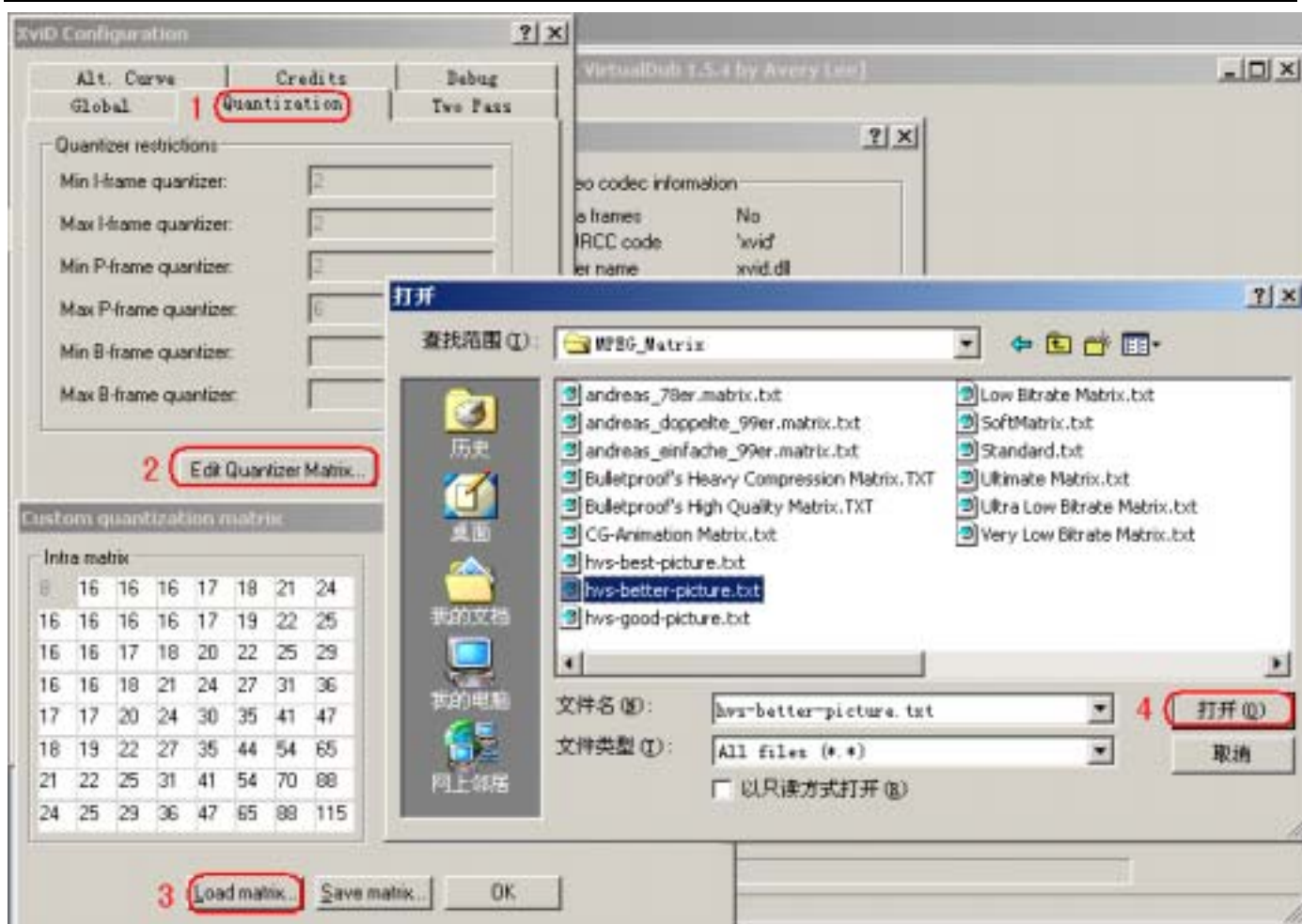


图 22

3) FourCC used: 改变成其他的 FourCC 码, 可以使文件在播放时利用不同的解码滤镜, 不过也可能造成播放错误。这里都用 XVID。

4) VHQ mode: 有 0-4 可选, 值越大, 搜寻精度更高, 效果越好, 速度越慢。如果机器够劲, 可以用 4 不计速度来求质量。注意: VHQ 2 和 VHQ 3 都会明显降低画质的, 所以不要用, 为了兼顾画质和速度使用 VHQ 1 是最佳的选择。

5) Maximum I-frame interval(最大关键帧距): 一般的理解最大关键帧距一般应该为 10-12 倍的每秒帧数(FPS)。比如 NTSC 23.976fps 就设 240, PAL 25fps 就设 250。如果想人为地多插入些关键帧的话, 这里可以设小点, 比如 120, 但相对会增加一定的压缩难度。

6) Minimum I-frame interval(最小关键帧距): 设置 1

7) 下面的 6 个选项, 只用 Use chroma motio 就可以了。Lumi masking 和 Quarterpel 很多人试验的结果都是弊大于利。

8) B-frame control:

Maximum B-frame: 可设 -1 和 2、3。-1 就是不用 BF, 对于 2CD、视频码率够大 (1100kbps 以上) 动态画面不多可以用 -1, 取消 BF, 只用 IF 和 PF 来压, 压出来质量比用 BF 好些。除上述情况外均应设 2 或 3。如果设 2, 就是指在两个 PF (也有可能是 IF 和 PF) 之间连续插入了 2 帧 BF (帧序: ...IF/PF **BF BF** PF...), 如果设 3, 就是指在两个 PF (也有可能是 IF 和 PF) 之间连续插入了 3 帧 BF (帧序: ...IF/PF **BF BF BF** PF...), 下面还有三个 BF 的控制值 (B-frame quantizer ratio(%), B-frame quantizer offset, B-frame threshold), 以 X、Y、Z 来代表, X 是倍数, Y 是偏差, Z 表示插入的频率, XviD 是智能判断是否需要插入 BF 的, 这个值控制判断的限度, 值越大, 插的越多。BF 的 Q 值由 BF 前后的 PF 的 Q 值的平均值来决定, 计算公式:

$$\text{BF 的 Q 值} = [\text{BF 前后的 PF (或者是 IF 和 PF) 的 Q 值的算术平均值} * X + Y] / 100$$

例 1: 前后 PF 的 Q 值分别为 3 和 4, BF 设置 X=100 Y=75, 则 BF 的 Q 值 = $\{[(3+4)/2]*100+75\}/100 = 4.25$ (理论值)

例 2: 前 IF 的 Q 值为 2, 后 PF 的 Q 值为 4, BF 设置 X=100 Y=200, 则 BF 的 Q 值 = $\{[(2+4)/2]*100+200\}/100 = 5$ (理论值)



图 23

根据实际应用的经验：如果比较难压的片子（不加任何降噪滤镜、使用 MPEG 量化的情况下，2pass/1pass size 比率 < 55%），推荐设置参数 2,100,75,50，就能使得 1pass size 有明显降低从而提高了 2pass/1pass size 比率。如果仍不理想，则视 1pass Size 减小的程度依次调整为 2,150,75,50 / 3,100,75,50 / 3,150,75,50（最大设置极限，再大的话则会影响画质了）。一般当 2pass/1pass size 比率高于 65% 就基本上可以达到比较满意的效果了。

特别提醒：

- 1) 不到万不得已的情况，BF 最大连续帧设置最好不要用 3；
- 2) BF 系列参数的设置应配合 Matrix 量化以及降噪滤镜来进行，就是说如果使用了 Matrix 量化和降噪滤镜而未用 BF 就可以将比率提高到 65% 以上时，则可不用设置 BF 参数；否则就要进行 BF 参数设置；
- 3) 如果比率提得过高（大于 100%）则最终的 2pass size 就会小于原来设定的容量值，而导致 undersize。

Two Pass:

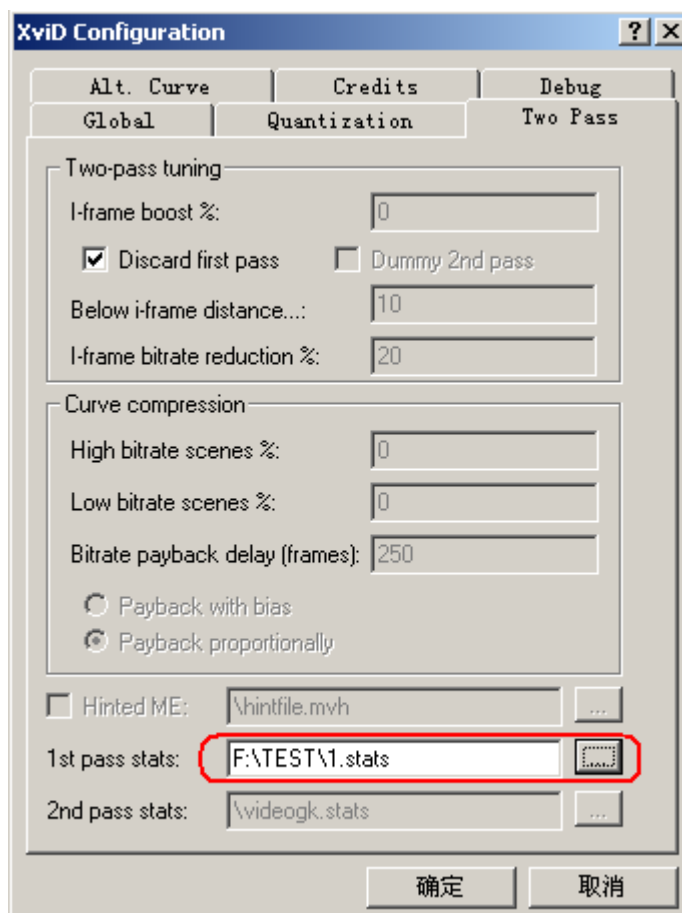


图 24

这个很简单了，通通默认就可以了。因为 Koepe 现在默认采用的都是 linear scale 的压缩方式，很多选项实际没有用处。比如 I-frame boost、Curve compression、Paybak with bias or proportionally 等，只要不采用 Curve 压缩，都无实际用处了。最后选择完成 1pass 过程产生的 stats 文件保存的路径即可。

Credits(演职员表):

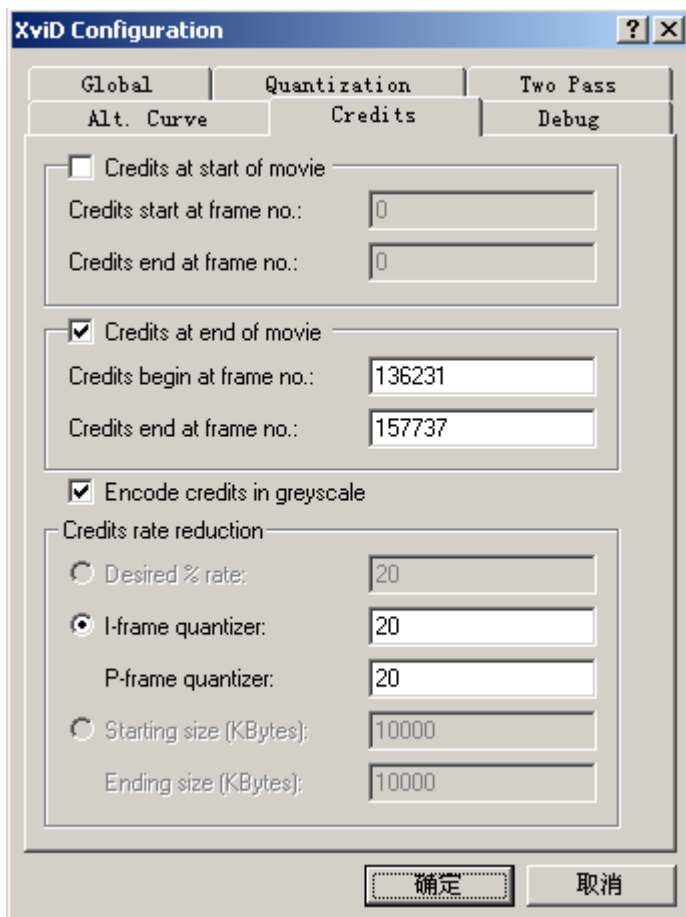


图 25

由于 Credits 部分通常都是滚动的字幕，类似这种滚动的画面或背景经过实践测试表明消耗的视频码率非常高（高达约 1000 ~ 2000kbps），浪费惊人，因此我们通过 Credits 设置选项，把演职员表等影片附属部分用很低的码率来压缩（即用较高的 I/PF Quant 值量化每个帧），节约出来的码率分配给正片。我个人习惯当全片码率比较高时（大于 1100kbps），可不设此选项。

具体设置方法：Xvid 提供了两段可以压缩的 Credits 设置，一般电影只压缩最后的演职员表，勾选"Credits at end if movie"后在"Credits begin at frame no:"处填入演职员表的起始帧，"Credits end at frame no:"处填入电影的结束帧。如果最后的 Credits 是黑白画面，还应该把下面的"Encode credits in greyscale"勾选上，以灰度模式来压，更加降低码率。下面的"Credits rate reduction"一般默认 20 就可以了，相对于强制这段时间的 I/PF 帧的压缩倍数（Quant）为 20。

Alt.Curve:

1pass 和 2pass 阶段都默认。

Debug:

1pass 和 2pass 阶段都默认。

至此，VDM 压缩的 1st pass 设置完成。

2、关于 2nd pass 设置:

首先 Encoding Mode 选择 2 Pass - 2nd pass Int，然后 Desired size 输入之前 GK 计算出的视频部分的容量，点击 Advanced options...进入 2pass 设置：

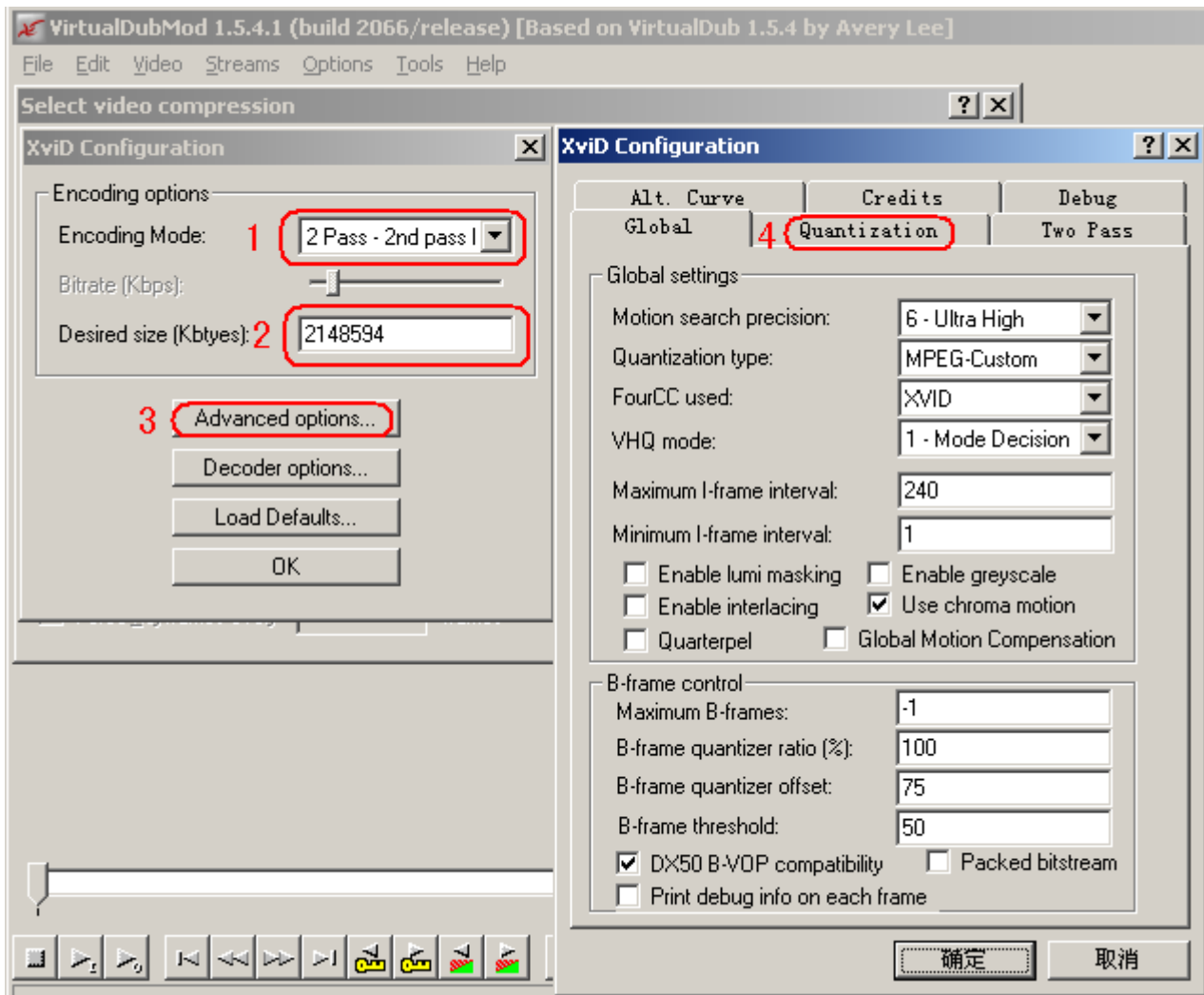


图 26

因为 2pass 基本上都是继承了 1pass 阶段的设置参数，所以通常只需设置 Quantization 中的项目即可，但如果 1pass 设置的是 H.263 量化方式，则还需对 Debug 中的项目进行设置。

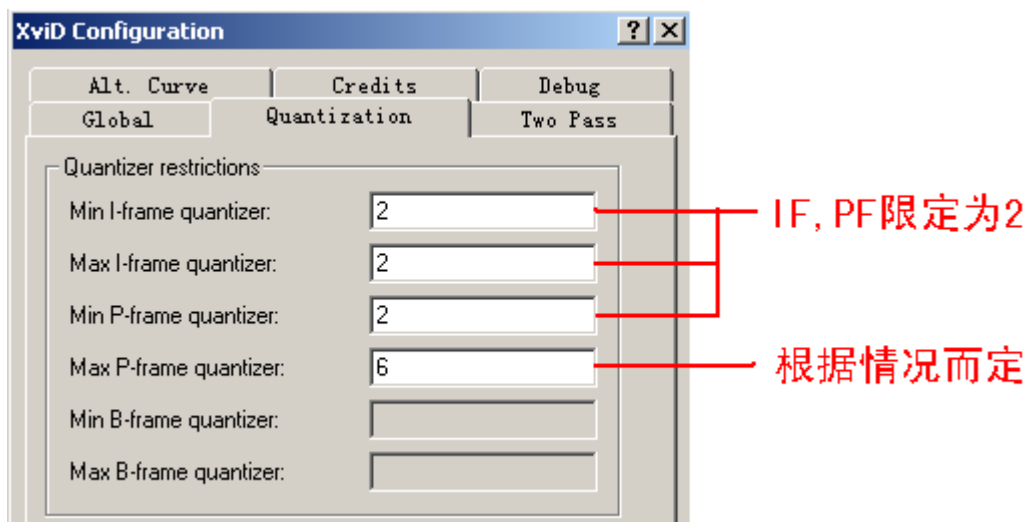


图 27

Quantization: 设置 IF 的最小和最大 Quant 值, PF 的最小和最大 Quant 值。因为从前面的 IF 和 PF 原理来看, IF 的 Quant 值是决定全片画质的最为关键因素。所以要限定 IF 的 Quant 值始终保持为 2, 但为了防止 2pass size 大于设定值 (oversize) 的情况, 所以 PF 的最大 Quant 值要根据 2pass/1pass size 比率情况进行限定或者不限定 (保持缺省的 31 值)。而具体限定为多少并没有定论, 一般以我的经验如果比率高过 65% 可限定为 8, 更高的话可以限定到 6 就基本上可以了, 不要再低了, 否则只会适得其反。



图 28

至此, 2nd pass 阶段的设置就算完成了。

各位同学, 看完了非常重要的 VDM 基于 Xvid 的 1pass、2pass 设置原理和经验之谈后, 马上回到我们的主题 - - AVS 脚本测试过程上来。导入 AVS 后, 首先设置好 1pass 参数, 然后回到 VDM 主界面在 Video->选择 Fast compress 压缩方式 (因为我们使用的是 YV12 编码方式所以必须选择此项)。接着 File->Save As...

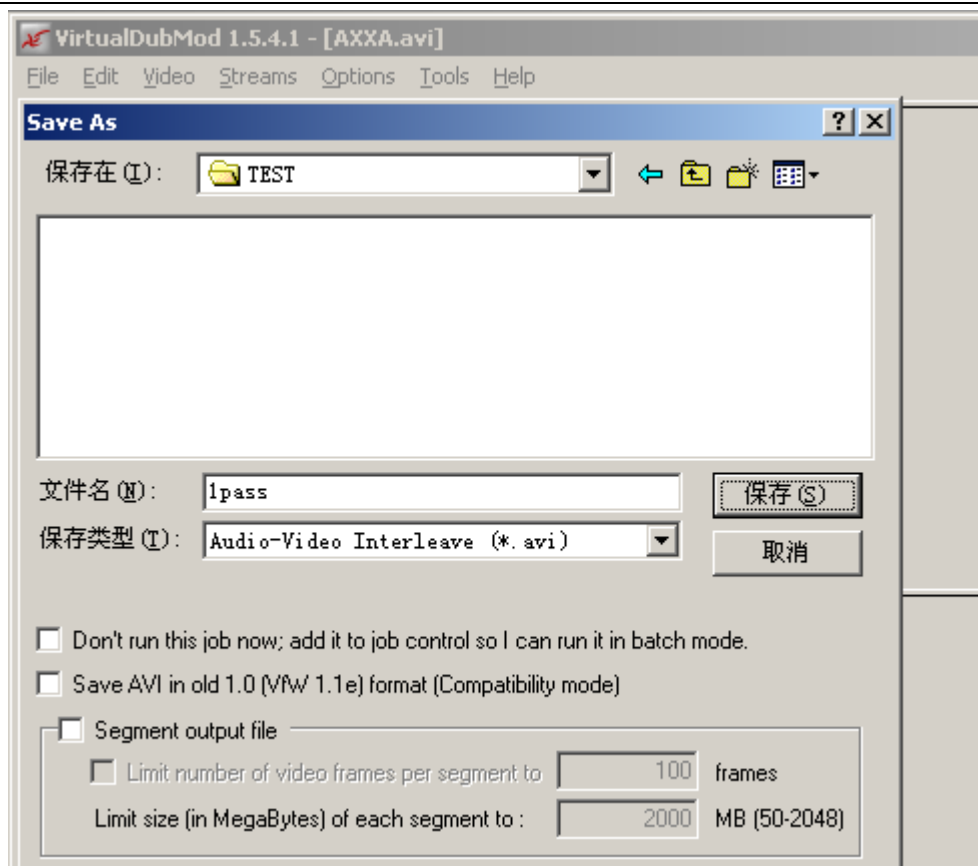


图 29

确定后,即开始 全片 5%平均采样的 AVS 脚本测试,之后会生成 1pass.avi 临时文件和 stats 文件 (后缀为 stats)。利用 XviD Stats Viewer 1.65 进行 1st pass 量化容量的查看。

运行 XviD Stats Viewer, 点击 Load Stats 加载 stats 文件, 从 Total Video Size 框内即可查看 1pass 量化的容量 (单位: KB)。则 5%AVS 脚本测试所得的 $2pass/1pass\ size = [2pass\ size / (Total\ Video\ Size / 5\%)] * 100\%$, 因为只是进行了全片 5% 的测试, 所以这个比率并不十分精准。但大量实践经验表明 5% 测试值通常要比 100% 测试值低 5% ~ 15% (例如, 5% 测试值为 55%, 那么 100% 测试值就有可能为 60% ~ 70% 左右。我的机器比较烂经常为了赶时间所以总偷懒。汗~~)。总之, 如果你的 CPU 够强劲还是建议使用全片 100% 的完整 1pass 测试, 方法如下:

```
SetWorkingDir("d:\GORDIA~1\")
LoadPlugin("mpeg2dec3.dll")
LoadPlugin("decomb.dll")
mpeg2source("F:\temp1\1.d2v")
telecide(guide=1)
Decimate(Cycle=5)
crop(14,8,-12,-12)
LanczosResize(640,352)
#SelectRangeEvery(280,14) 将此项用 # 字符注释
```

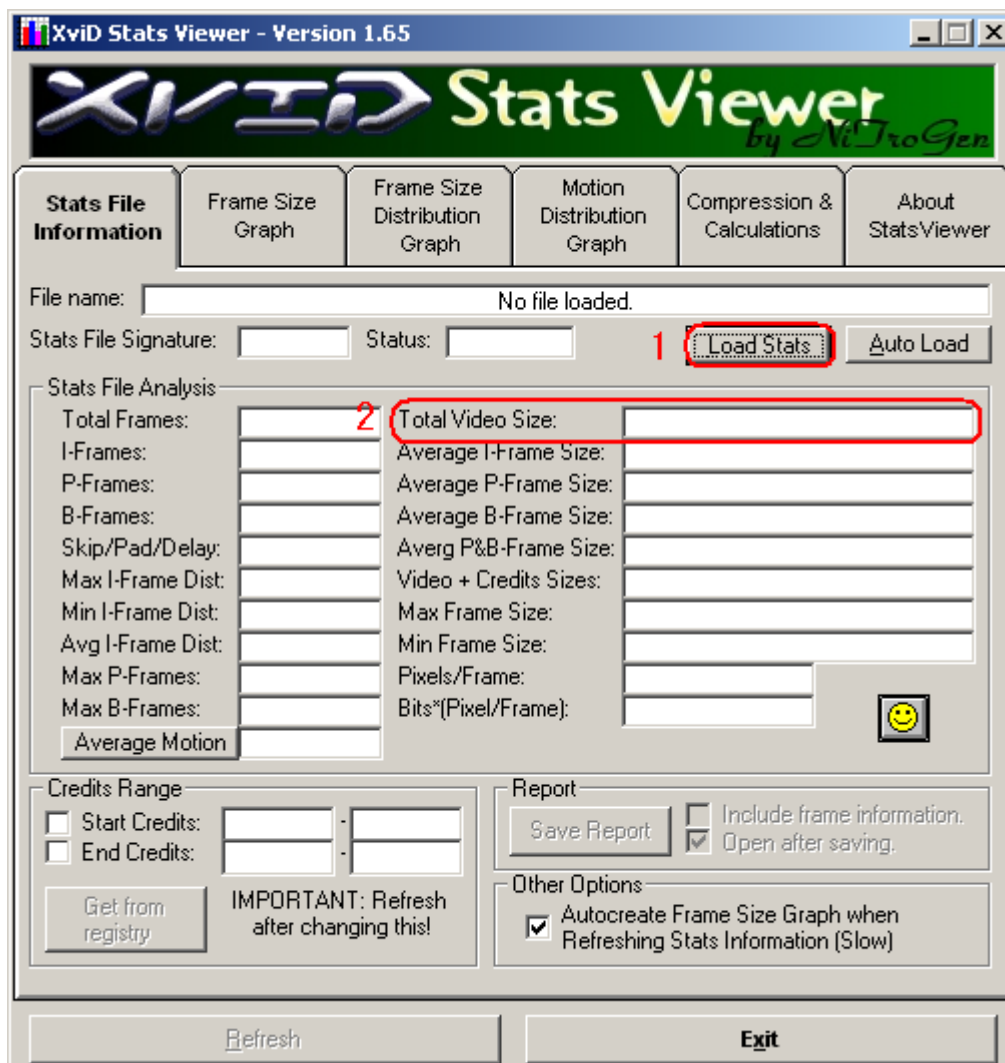


图 30

影响 1pass size 的因素大致有以下几个方面

1) 片源噪点

即影片的亮斑点和色斑点，由胶片本身的感光颗粒形成，老片更是如此，由于是不停闪烁的所以它们所消耗视频码率不可小视。解决办法就是在 AVS 脚本文件中加入降噪滤镜，平滑这些亮斑点和色斑点，使得噪点的 亮度 和 色度 均匀平衡，自然就消除了闪烁感，从而有效地抑止了噪点对码率的消耗。

普通影片一般使用 undot、Convolution3dYV12、MipSmooth 等，动画影片一般使用 undot、MipSmooth、deen、edeen 等（其实动画的滤镜运用要比普通影片的滤镜丰富的多，由于本人水平所限，暂时只能列举这些了），视噪点面积和密度来决定使用何种降噪滤镜。

由于降噪滤镜是一把双刃剑，它们是通过牺牲部分画面细节来换取整洁干净画面，降噪后会产生一定的画面轮廓扩散，所以要使用 MSharpen 滤镜进行边缘锐化。具体的使用方法和经验见“NewMov DivX 小组工作区”的相关讨论帖子和滤镜插件包内的 readme 文件。

以下就以普通影片加入降噪滤镜的 AVS 脚本进行简要举例说明：

```
SetWorkingDir("d:\GORDIA~1")
LoadPlugin("mpeg2dec3.dll")
LoadPlugin("decomb.dll")
LoadPlugin("Undot.dll")           # 加载 undot.dll 滤镜插件
LoadPlugin("Convolution3dYV12.dll") # 加载 Convolution3dYV12.dll 滤镜插件
LoadPlugin("MSharpen.dll")        # 加载 MSharpen.dll 滤镜插件
mpeg2source("F:\temp1\1.d2v")
telecide(guide=1).Decimate(Cycle=5)
```

```

crop(14,8,-12,-12)
Convolution3d(preset="MovieLQ")      # 调用 C3D 使用 MovieLQ 降噪预置函数
undot()                              # 调用 undot 滤镜进行像素级细小噪点降噪
LanczosResize(640,352)
undot()                              # 对变形后画面的像素级噪点进一步降噪
MSharpen(4,30,true,false,false)     # 收敛经过降噪后引起的画面轮廓扩散

```

2) 变形分辨率

道理很简单，最终的分辨率越大，所用的像素点就越多，自然容量也越大了。不过我这里需要说明的是，大家千万不要为了降低 1pass size 而极端地降低分辨率。因为太低的分辨率画面清晰度和细节表现都会受到影响，一般纵横比为 1:1.85 的影片的分辨率宽度不要低于 560 个像素，纵横比为 1:1.33 的影片的分辨率宽度不要低于 512 个像素。

3) 1pass 量化方式

这个在前面的 1st pass 设置原理中已经做过对比，这里就不再赘述。

4) BF 运用

因为影片的压缩率是在 1pass 阶段确定下来的，所以通过在两个 PF（或者 IF 和 PF）之间插入压缩率较高（即 Q 值较高）的 BF 来降低 IF 和 PF 的压缩率（即降低 Q 值），以提高整片的压缩率，达到降低 1pass size 的目的。Q 值可以调高是因为 BF 的信息含量比较少，解码时还可以参考前后帧的数据，这样 BF 就可以压缩得狠一点，所以采取 BF 对付有压缩难度影片是非常奏效的。当然，加 BF 也得有限度，具体到是否加？以及如何设置？我在前面的章节已做了比较详尽的说明，这里也就不再赘述。

以上几个方面都是影响 1pass size 的主要因素，在这些因素里找平衡才是我们要去学习、研究和大量实践的。通过 AVS 脚本测试，找出最佳的压缩方案（即 1pass 设置参数、AVS 脚本内容）。

第四节 整片压缩

1、根据确定好的 AVS 脚本文件，导入 VDM 进行 1pass 和 2pass 一次性 Job 方式压缩。

1) 在 1pass 设置完成后，依照下图保存 1pass.avi：

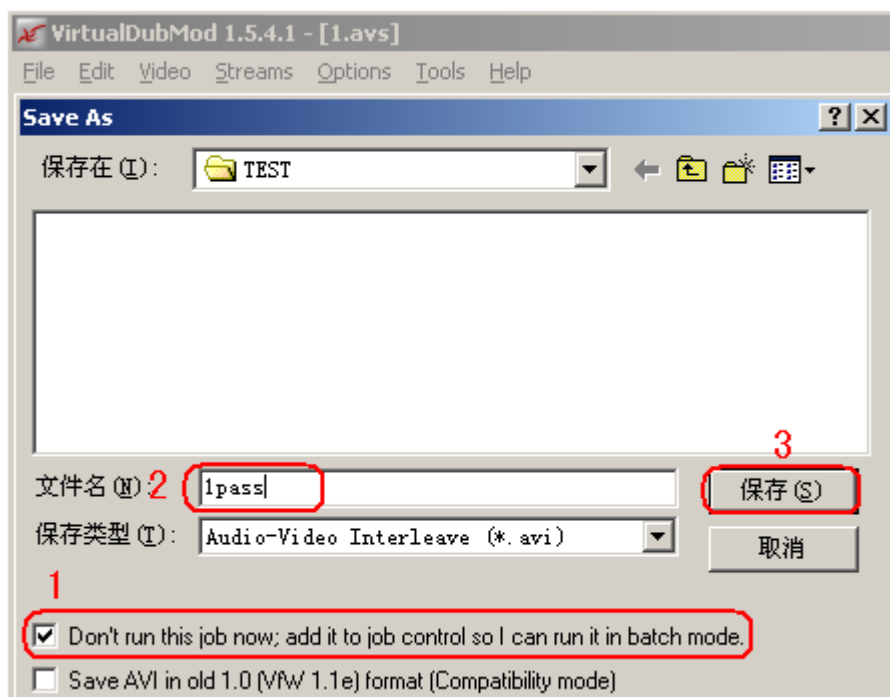


图 31

2)不要关闭 VDM ,接着进行 2pass 设置 ,然后按照相同方法保存 2pass.avi ,最后在到 File 菜单选择 Job control :

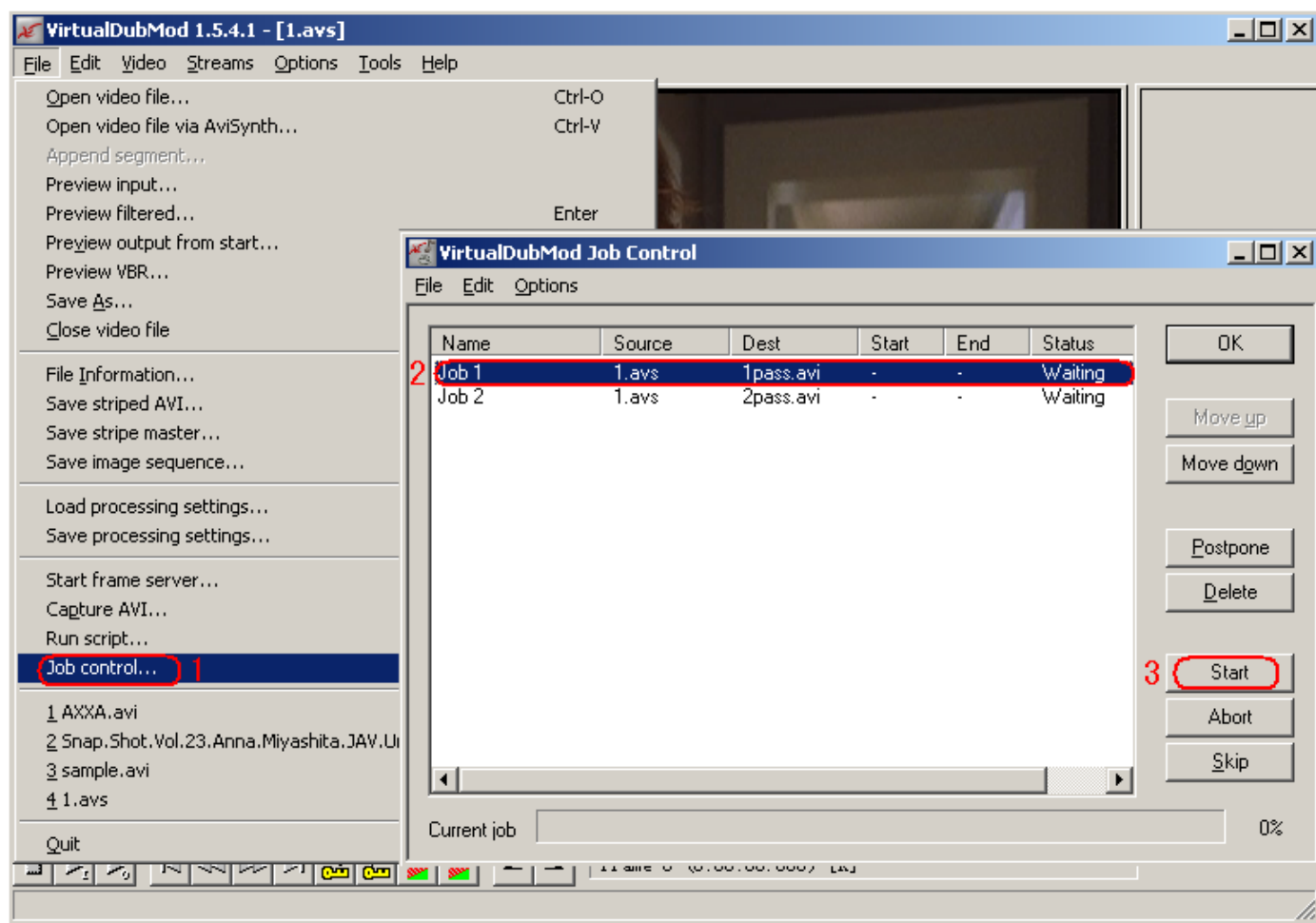


图 32

由于我们选择了 Job control.....所以上图你会看到所有的任务都已经安排好了你可以按 Start 开始运作或者在 File 里头保存一份清单,想在那时候做就看你决定了,这里要有心里准备,因为要花的时间比较长的。Job 完成后不含音频的 2pass AVI 文件就诞生了。

第五节 音频格式转换

为什么要做音频格式转换的步骤? 主要是我们若做成 1CD 版本的 DVDrip 时搭配 AC3 音频的话我们都知道 AC3 音频的文件太大了,所以也表示你实际上处理视频的文件太少了,这样会造成质量降低码率偏低,所以我们就必须做音频格式转换的步骤,简单地说:降低音频文件等于提高视频文件的质量。

所以一般来说我们都是用 128k 的 MP3 来做成 1CD 版本的 DVDrip,这里我们可以使用 GK 来方便地完成 AC3 音频转成 MP3 的工作,做法是在 GK 界面中的 Encoder 里选取 Add Job 就出现。如下图所示:

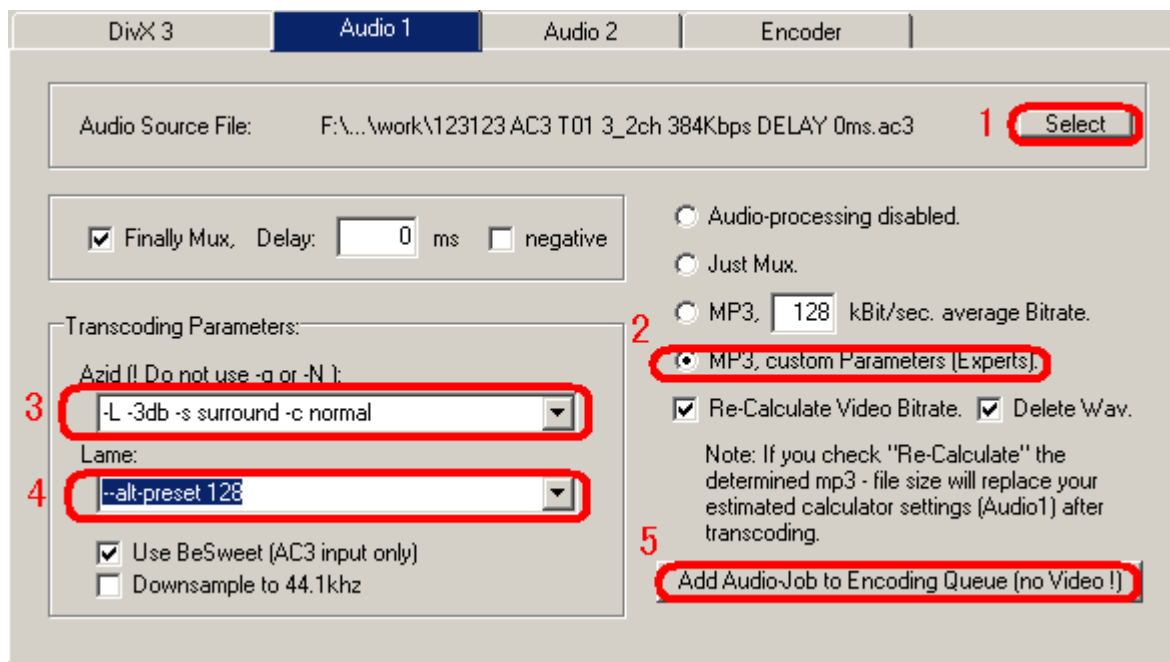


图 33

第六节 合并视频和音频

通常的办法是使用 Nandub 来完成此项工作,但这里我要介绍是一种超傻瓜的合成工具 - AVI-Mux_GUI 1.15.3,将 第四节 中生成的 2pass AVI 文件合上 AC3 或 MP3 音轨文件。具体操作步骤如下:



图 34

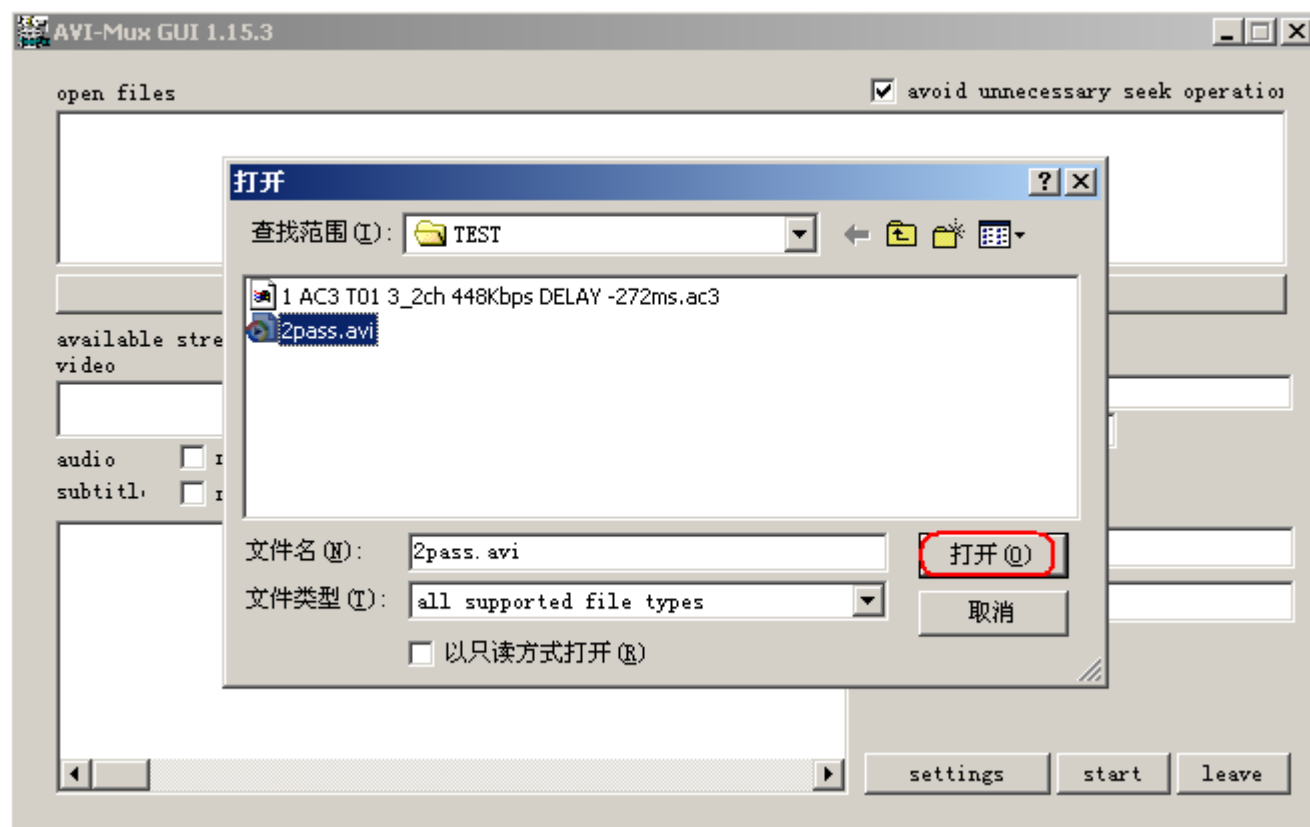


图 35

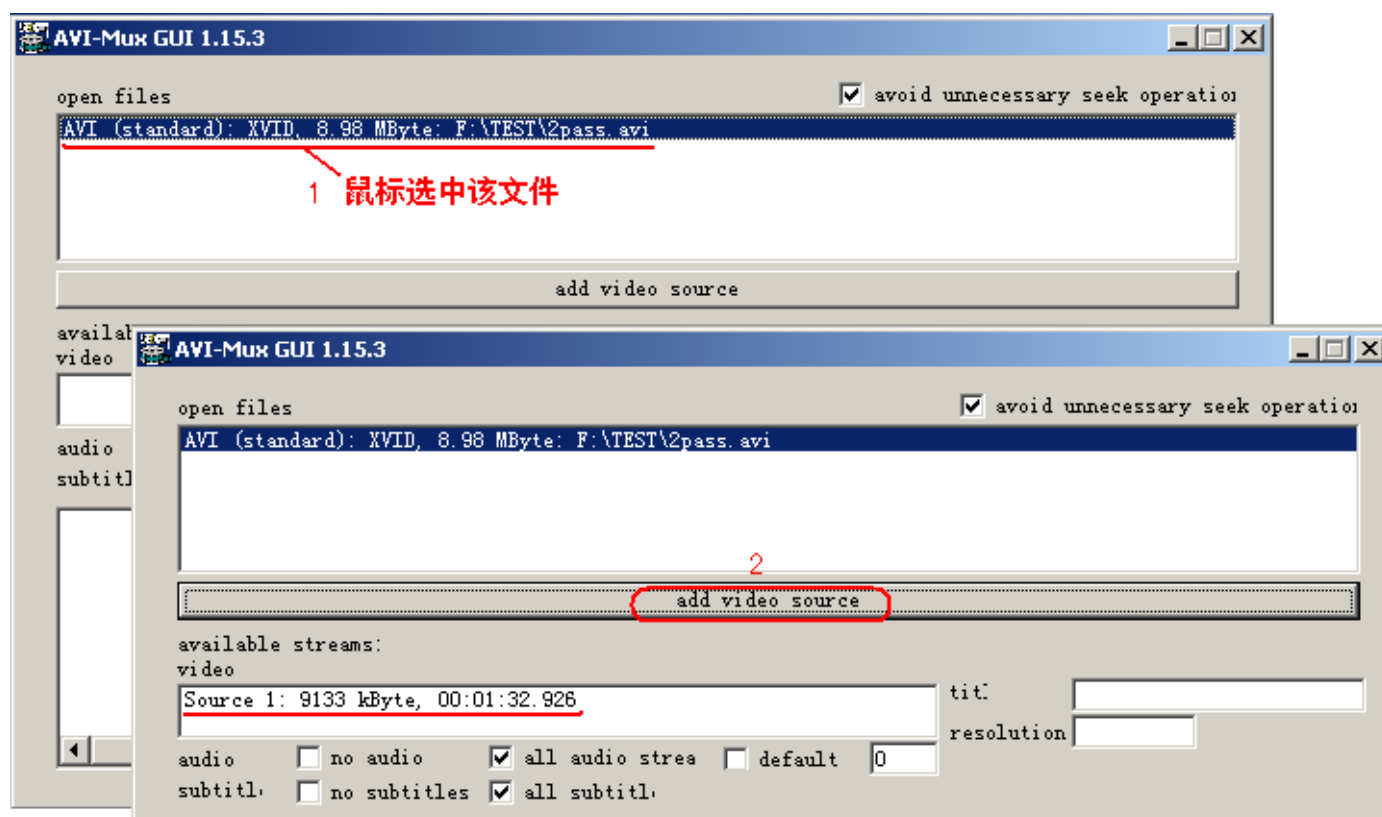


图 36

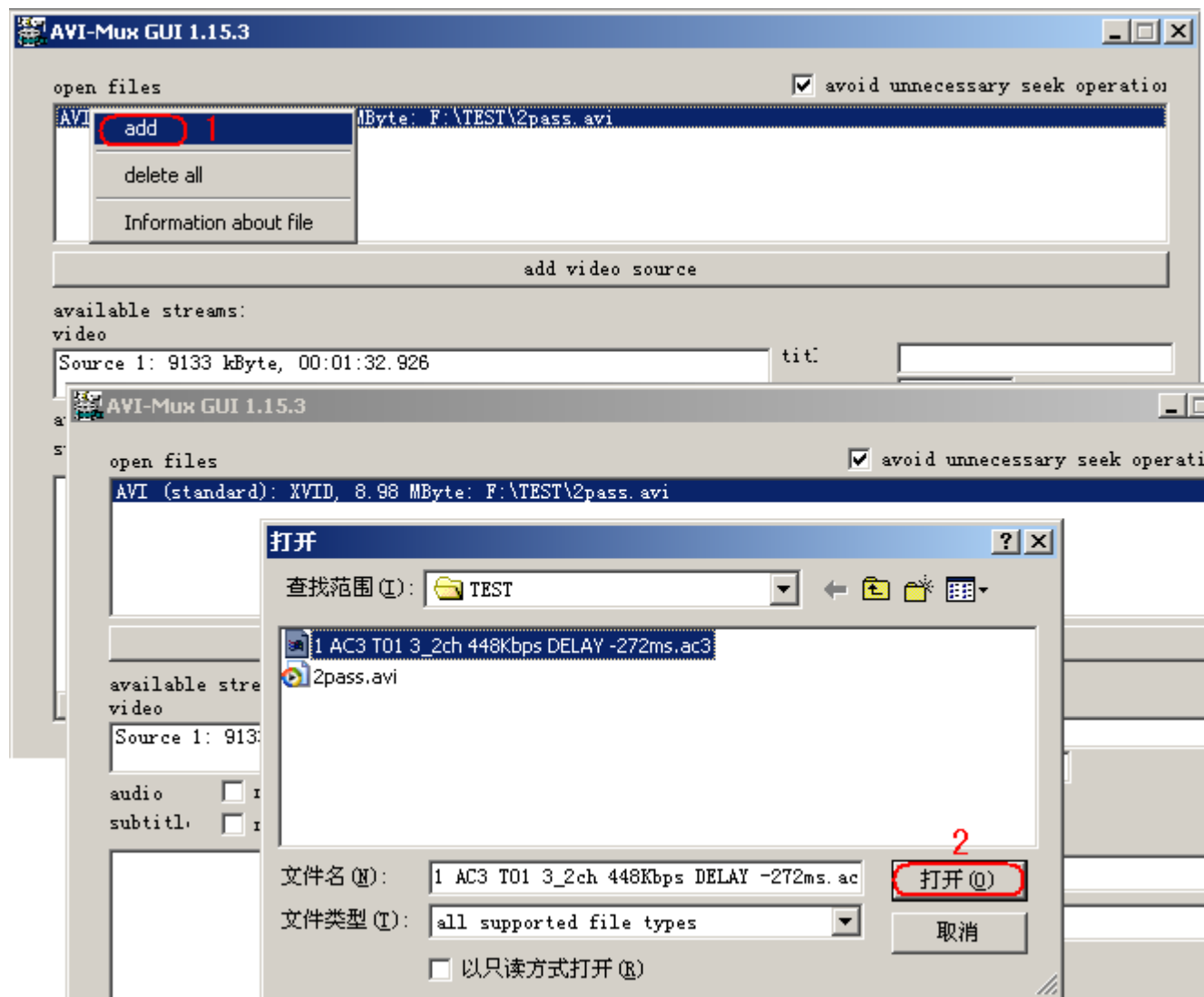


图 37

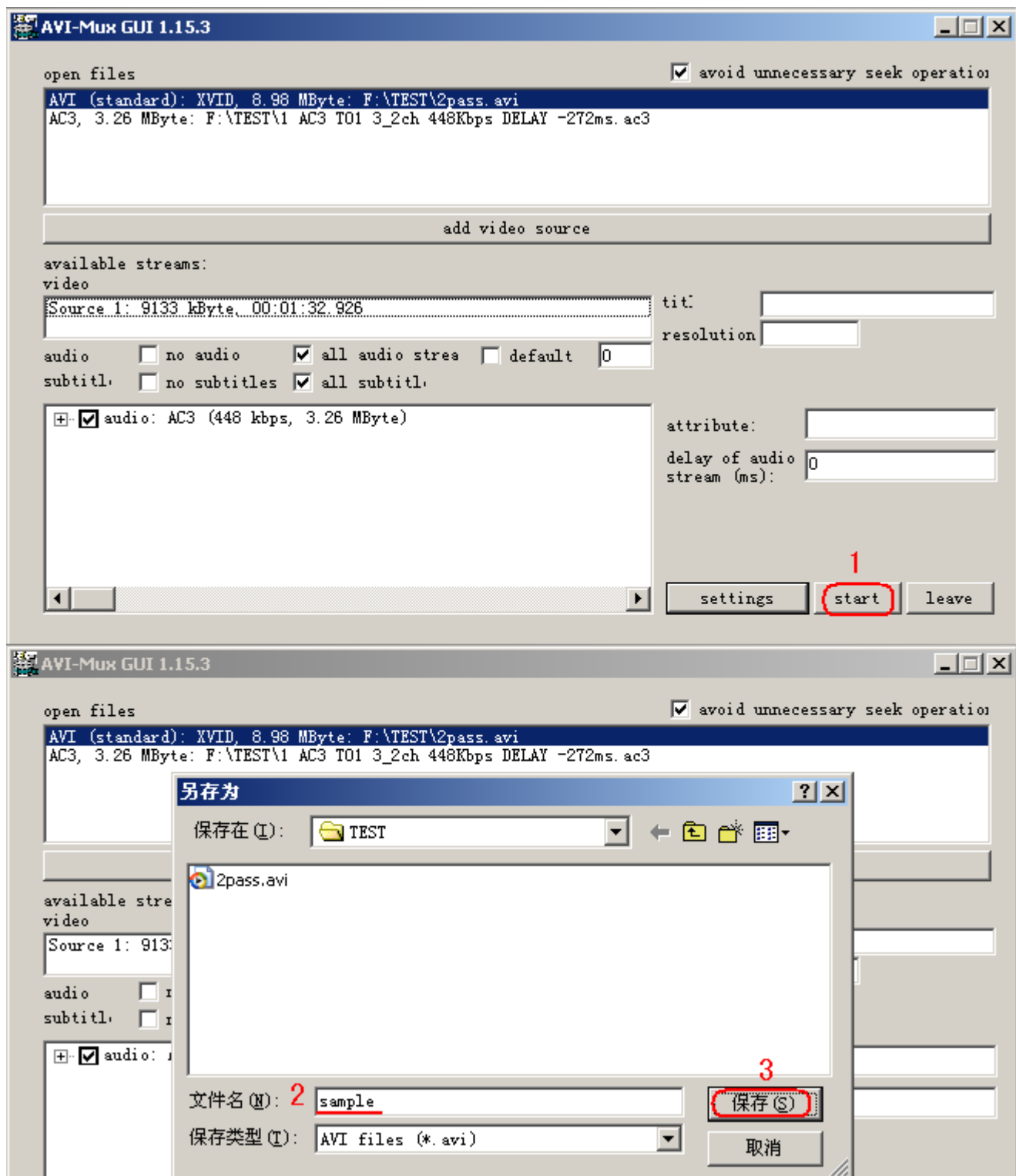


图 38

这样，一部完整的 AVI 文件就大功告成了。之所以说它傻瓜是因为整个合并过程完全使用默认设置即可，而且还支持 N 条音轨的合并，方法与单音轨合并方法同理。

第七节 提取字幕

提取字幕可以在 GK 里做。在 subtitles-->Configure-->Open...这里我们选择我们之前拷贝到硬盘里的*.IFO 文件。

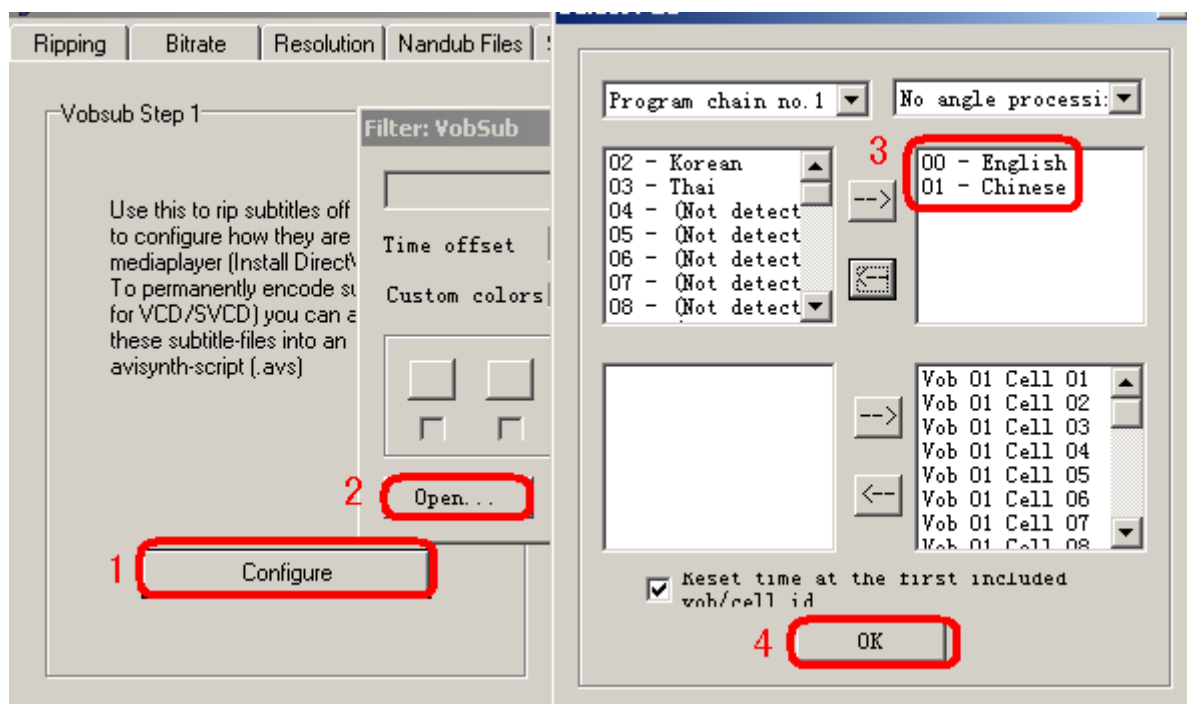


图 39

按 ok 就开始 rip 字幕了..... 等待工作完后你在选择 English 为第一语言就行了。选择 English 为第一语言表示每次看影片时加载的字幕都是英文为优先，其次是中文，点击 OK 生成 *.idx 和 *.sub 字幕文件到指定的路径。

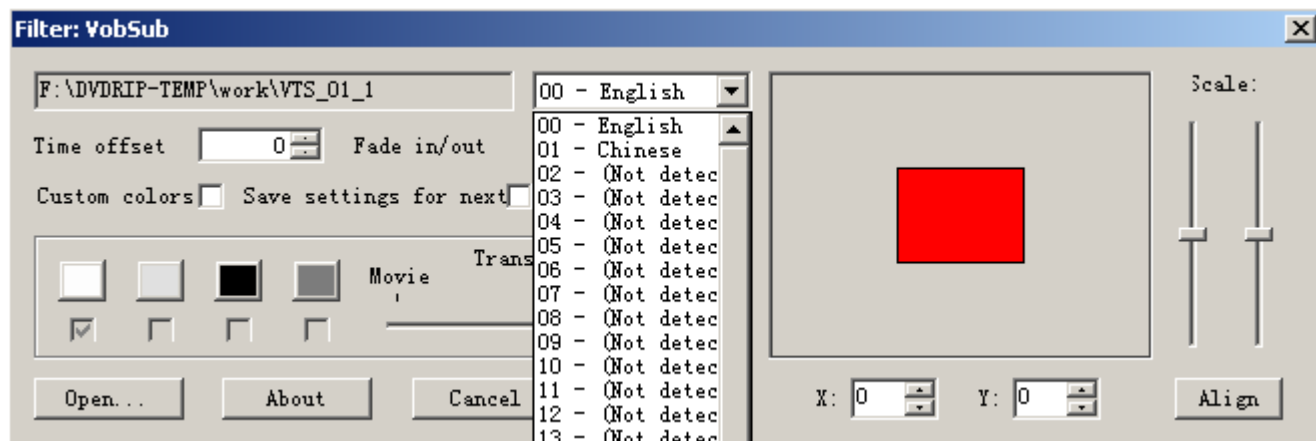


图 40

(全文完)

后话：至于字幕分割、AVI 文件分割、Gspot 和 DRFAnalyzer 等工具的用法在“NewMov DivX 小组工作区”和射手网都有详细的教程，这里我就不多说了。哈哈~~~又偷懒了。。:-)

(特别鸣谢：本篇文章部分理论和描述引自 Doom9.org 以及 net1999、csr2000、fantasy、rkingbo、wen7773、nanji、加林仙豆等网友的相关文章和技术支持)