

Autodesk® Scaleform®

CRI 视频编码器命令行工具 概述

本文件描述了 CRI 编码器命令行工具在 Scaleform Video 中的应用。

编者: Vladislav Merker

版本: 1.02

最后修订: 2013 年 1 月 18 日

Copyright Notice

Autodesk® Scaleform® 4.3

© 2013 Autodesk, Inc. All rights reserved. Except as otherwise permitted by Autodesk, Inc., this publication, or parts thereof, may not be reproduced in any form, by any method, for any purpose.

Certain materials included in this publication are reprinted with the permission of the copyright holder.

The following are registered trademarks or trademarks of Autodesk, Inc., and/or its subsidiaries and/or affiliates in the USA and other countries: 123D, 3ds Max, Algor, Alias, AliasStudio, ATC, AutoCAD, AutoCAD Learning Assistance, AutoCAD LT, AutoCAD Simulator, AutoCAD SQL Extension, AutoCAD SQL Interface, Autodesk, Autodesk 123D, Autodesk Homestyler, Autodesk Intent, Autodesk Inventor, Autodesk MapGuide, Autodesk Streamline, AutoLISP, AutoSketch, AutoSnap, AutoTrack, Backburner, Backdraft, Beast, Beast (design/logo), BIM 360, Built with ObjectARX (design/logo), Burn, Buzzsaw, CADmep, CAiCE, CAMduct, CFdesign, Civil 3D, Cleaner, Cleaner Central, ClearScale, Colour Warper, Combustion, Communication Specification, Constructware, Content Explorer, Creative Bridge, Dancing Baby (image), DesignCenter, Design Doctor, Designer's Toolkit, DesignKids, DesignProf, Design Server, DesignStudio, Design Web Format, Discreet, DWF, DWG, DWG (design/logo), DWG Extreme, DWG TrueConvert, DWG TrueView, DWGX, DXF, Ecotect, ESTmep, Evolver, Exposure, Extending the Design Team, FABmep, Face Robot, FBX, Fempro, Fire, Flame, Flare, Flint, FMDesktop, ForceEffect, Freewheel, GDX Driver, Glue, Green Building Studio, Heads-up Design, Heidi, Homestyler, HumanIK, i-drop, ImageModeler, iMOUT, Incinerator, Inferno, Instructables, Instructables (stylized robot design/logo), Inventor, Inventor LT, Kynapse, Kynogon, LandXplorer, Lustre, Map It, Build It, Use It, MatchMover, Maya, Mechanical Desktop, MIMI, Moldflow, Moldflow Plastics Advisers, Moldflow Plastics Insight, Moondust, MotionBuilder, Movimento, MPA, MPA (design/logo), MPI (design/logo), MPX, MPX (design/logo), Mudbox, Multi-Master Editing, Navisworks, ObjectARX, ObjectDBX, Opticore, Pipeplus, Pixlr, Pixlr-o-matic, PolarSnap, Powered with Autodesk Technology, Productstream, ProMaterials, RasterDWG, RealDWG, Real-time Roto, Recognize, Render Queue, Retimer, Reveal, Revit, Revit LT, RiverCAD, Robot, Scaleform, Scaleform GFx, Showcase, Show Me, ShowMotion, SketchBook, Smoke, Softimage, Socialcam, Sparks, SteeringWheels, Stitcher, Stone, StormNET, TinkerBox, ToolClip, Topobase, Toxik, TrustedDWG, T-Splines, U-Vis, ViewCube, Visual, Visual LISP, Vtour, WaterNetworks, Wire, Wiretap, WiretapCentral, XSI.

All other brand names, product names or trademarks belong to their respective holders.

Disclaimer

THIS PUBLICATION AND THE INFORMATION CONTAINED HEREIN IS MADE AVAILABLE BY AUTODESK, INC. "AS IS." AUTODESK, INC. DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE REGARDING THESE MATERIALS.

如何与 Autodesk Scaleform 联系:

文件名称	CRI 视频编码器命令行工具概述
地址	美国格林贝尔特 MD 20770 常春藤路 6305 号 310 室
	Scaleform 公司
网站	www.scaleform.com
电子邮件	info@scaleform.com
直线电话	(301) 446-3200
传真	(301) 446-3199

目 录

1 概述	1
2 输入/输出文件	1
2.1 视频素材文件.....	1
2.1.1 视频素材规格.....	1
2.1.2 顺序编号的静态图像文件规格.....	2
2.2 音频素材文件.....	3
2.3 字幕信息文件.....	3
2.3.1 字幕信息文件的格式.....	3
2.3.2 字幕信息文件示例	4
2.4 提示点信息文件	4
2.4.1 提示点信息文件的格式.....	4
2.4.2 提示点信息文件示例	5
3 CRI 编码器的基本用法	6
3.1 子命令	6
3.1.1 文件子命令	6
3.1.2 Video 视频编辑子命令	6
3.1.3 编码设置子命令	7
3.1.4 音频编码设置子命令	7
3.1.5 杂项子命令	7
3.2 子命令规格	7
3.2.1 文件子命令	7
3.2.2 视频编辑子命令	11
3.2.3 Video 编码设置子命令	11
3.2.4 GOP (图片组)子命令	13
3.2.5 音频编码设置子命令	14

1 概述

The Scaleform® Video™ 解决方案由 CRI Movie™ 编码器驱动，在个人电脑和游戏机上播放高质量视频。在播放视频文件之前，必须通过 Scaleform 编码器工具将 AVI 视频转换为 USM 格式，因为 Scaleform Video 只能处理以 CRI 格式载入的外部文件。CRI 视频编码器在游戏机上的应用版本是一个命令行工具，它能创建 CRI 视频数据。本文件描述了如何使用这一工具，以及在解码过程中需要什么参数。

2 输入/输出文件

下面列出了用于解码的输入文件和所生成的输出文件。

- **输入文件**
 - 视频素材文件
 - 音频素材文件
 - 字幕信息文件
 - 提示点信息文件
- **输出文件**
 - CRI 视频数据 (.usm)

2.1 视频素材文件

2.1.1 视频素材规格

建议使用带有无压缩视频与音频轨道的 AVI 文件作为视频素材。此外，顺序编号的静态图像文件也可用作视频素材。

推荐的视频素材规格	
项目	描述
文件类型	Microsoft AVI (支持大于 2GB 的文件)。
音频轨道	无压缩 PCM 格式(等同于 WAVE)。
视频轨道	无压缩 RGB 格式(等同于 BMP)。

视频素材规格	
项目	描述
文件类型	AVI 文件 (*1). 顺序编号的静态图像文件
色深	24-位 / 32-位色
图像尺寸	[最小]72x72, [最大] 4088x4088, 像素区块 8x8 (*2).
帧速率	60 /59.94 /50 /30 /29.97 /25 /24 /23.976 帧/秒 (*3).
像素长宽比	[宽屏电视] 方形像素 (*4) [正常电视] D1 像素

[注释]

(*1) 这一工具使用 DirectShow 界面，因此，基本上能够处理微软媒体播放器所播放的任何 AVI 文件。但不担保能够对这些 AVI 文件全部进行正确解码。

某些 AVI 文件无法读取或者无法完全解码，这取决于它所使用的编码器（编码器必须完全遵守 DirectShow 过滤器的规格）。

(*2) 对于 PlayStation 2 视频来说，图像尺寸应当为 16x16 像素块。

(*3) 推荐使用的帧速率为 29.97 帧/秒。

尽管具有较低帧速率的视频素材不符合标准，但也能处理。

(*4) 为宽屏电视制作 Wii 视频时需要特别的视频素材。

2.1.2 顺序编号的静态图像文件规格

顺序编号的静态图像文件是指视频中每一帧按照顺序编号保存的静态图像文件。CRI 编码器工具可以处理以下规格的顺序编号静态图像文件：

视频素材规格	
项目	描述
格式	1) 24-位 / 32-位色 BMP (无压缩)。 2) 24-位 / 32-位色 TGA (无压缩 / 运行长度)。
图像尺寸	[最小]72x72, [最大] 没有限制 (8x8 像素块)。
文件命名规则	1) "基本名字" + "编号" + ".扩展名" 例如 OK> abc001.bmp, abc002.bmp, abc003.bmp, ... 例如 NG> abc001def.bmp, abc002def.bmp, abc003def.bmp, ... 2) 可使用进位 E.g., OK> abc998.bmp, abc999.bmp, abc1000.bmp, ...

	3) 丢失的文件将被跳过。 例如 OK> abc001.bmp, abc004.bmp, abc008.bmp, ...
--	---

2.2 音频素材文件

影片的音频素材可通过下列方式之一进行规定：

- 使用 AVI 文件中所包含的音频轨道，或者
- 将视频素材文件与音频素材文件分离。

下面给出音频素材的规格：

推荐的视频素材规格	
项目	描述
文件类型	WAVE 波形或 AIFF 格式
格式	16-位线性 PCM (无压缩).
频道编号	立体声或单声道
采样频率	最高支持 48kHz

2.3 字幕信息文件

如果影片中需要字幕，则应当编制字幕信息文件。这是一个用来描述何时显示字幕（字符串）的文本文件。

2.3.1 字幕信息文件的格式

1. 时间单位（整数）在第一行进行定义。
2. 在第二行或下面，定义了开始时间（INTIME）、结束时间（OUTTIME）和字幕内容，这些数据以“，（逗号）”隔开。
3. 开始时间（结束时间）的时间单位为妙。
4. 播放价格不包括在结束时间内。
5. ‘,’（逗号）在字幕内容中按一个字符处理。
6. 以‘;’（分号）开始的行当做评论处理。
7. 能够支持的字符编码为 ASCII、Shift JIS、UTF-8 和 UTF-16 (UTF-16LE)。
这些内容在输入影片中时不进行任何修改。

2.3.2 字幕信息文件示例

示例 1

字幕信息是以播放时间为基准进行书写的。

在这一示例中，时间单位为 1000。每一个字幕内容所显示的时间分别为 0-2 秒、5-7.5 秒以及 7.5-10 秒。

```
;INTIME, OUTTIME, Subtitle, Comment  
1000  
0, 2000, FirstMessage.  
5000, 7500, Second Message.  
7500, 10000, Third Message.
```

示例 2

字幕信息是以帧数为基础进行书写的。

例如，如果时间单位为 29970，帧速率为 29.97，则每一帧的播放间隔为 1000，播放的帧数为 $29.97 \times 1000 = 29970$ 。

在这个示例中，时间单位为 29970。每一个字幕内容所显示的时间分别为 0 – 100 帧、150 – 200 帧、和 200 – 250 帧。

```
;INTIME, OUTTIME, Subtitle, Comment  
29970  
0, 100000, FirstMessage.  
150000, 200000, Second Message.  
200000, 250000, Third Message.
```

2.4 提示点信息文件

“提示点”是在影片中放置时间信息（事件点）的一种功能。当在播放过程中达到每一个事件点的时间后，回调函数开始被调用。“提示点信息文件”是一个定义这些事件点的文本文件。

2.4.1 提示点信息文件的格式

1. 时间单位（整数）在第一行进行定义。
2. 在第二行或下面，“Time”、“Value”、“EventPointName”和“String”在同一行进行定义。这些数据以 ‘,’ (逗号) 隔开。

3. 第一个参数"Time"规定了事件发布的时间， "Time"的时间单位为妙。
4. 第二个参数 "Value" 是一个能够在程序中使用的数值。
5. 第三个参数 "EventPointName" 为该事件点的名字。
6. 第四个参数"String"是一个能够在程序中使用的字符串。程序库不会对这一参数进行分析，它应当由程序来处理。
7. 以 ';' (分号)开始的行按评论处理。
8. 支持的字符编码为 ASCII、 Shift JIS、 UTF-8 和 UTF-16 (UTF-16LE)。

2.4.2 提示点信息文件示例

示例 1

在本示例中，时间单位为 1000。

事件分别在 0、5、7、8、15 和 25 秒进行发布。

事件点 #3 和 #4 的字符串通过逗号隔开。他们应当在应用程序中进行分析。

```
;Time, Value, EventPointName, String
1000
0,0,navi1
5000,0,navi2
7000,1,evpt1,param,test
8000,1,evpt2,param1,300,param2,500
15000,0,navi3
15000,1,evpt3
25000,0,navi4
```

3 CRI 编码器的基本用法

在启动程序时，打开 MS-DOS 命令提示符，并按照下列格式输入命令。

```
[Format] medianoche [-subcommand=param1[,param2,...]]  
[Input]   subcommand : subcommand  
           param#    : subcommand parameter
```

示例：

通过对输入的 AVI 文件 "sample.avi" 进行编码而创建一个 CRI 影片文件 "sample.usm"。

```
> medianoche -in=sample.avi -out=sample.usm
```

3.1 子命令

“子命令”是一个命令选项，它会告知程序应当执行什么进程。

例如，它规定了输入文件、输出文件和运行模式。

3.1.1 文件子命令

子命令	描述
in	规定输入视频文件
out	规定输出视频文件
video##	规定输入视频文件
alpha##	规定 Alpha 文件
audio##	规定音频文件
mca**_##	为环绕影片规定音频文件
subtitle##	规定字幕信息文件
cuepoint	规定提示点信息文件

3.1.2 Video 视频编辑子命令

子命令	描述
crop	对原图像进行剪切
scale	对原图像进行放大或缩小

3.1.3 编码设置子命令

子命令	描述
bitrate	比特率 (位/秒)。
br_range	可变比特率控制的比特率范围
framerate	帧速率
hcfilter	高压缩过滤器
gop_i	I 图片在图片组中的数量
gop_p	P-图片在图片组中的数量
gop_b	B-图片在图片组中的数量
gop_closed	关闭 GOP (图片组)
detectsc	场景变化检测
ms_fullpel_still	静态部分全像素动作查询

3.1.4 音频编码设置子命令

子命令	描述
hca	对 HCA 编解码器的音频材料进行编码。
hca_quality	作为 HCA 编码时的声音质量。

3.1.5 杂项子命令

子命令	描述
cleanup	删除中间文件
work_dir	设置工作目录
debug	输出调试信息
preview	显示预览屏幕

3.2 子命令规格

3.2.1 文件子命令

in: 规定输入视频文件

```
[Format]      -in=filename  
[Input]      filename : input video file  
[Function]   Specifies input video file.
```

这是一个常用的子命令，并提供了一个视频替代子命令。

示例：规定 sample.avi 作为一个输入视频文件。

```
> medianoche -in=sample.avi -out=sample.usm
```

out: 规定输出视频文件

```
[Format]      -out=filename  
[Input]      output : output movie file (usm file)  
[Function]   Specifies output file name.
```

这是一个常用子命令。规定输出文件的扩展名为"usm"。

示例：规定 sample.usm 作为输出文件名。

```
> medianoche -in=sample.avi -out=sample.usm
```

video: 规定输入视频文件

```
[Format]      -video##=filename  
[Input]      filename : video file  
            ##       : video track number (## = 00)  
[Function]   Specifies input video file.
```

1. 视频文件：有关视频文件的详细信息，请参阅第 2.1 部分内容。
2. 视频轨道编号：请保证将 00 作为视频轨道的编号。目前，不支持其他轨道编号。

示例：将一个 AVI 文件 smpvid.avi 分配给轨道 00。

```
> medianoche -video00=smpvid.avi -audio00=smpaud.wav -out=sample.usm
```

alpha: 规定 Alpha 文件

```
[Format]      -alpha##=filename  
[Input]      filename : alpha file  
            ##       : alpha track number (## = 00)  
[Function]   Specifies input alpha file.
```

在创建 alpha 视频时，规定一个包含 alpha 编码频道的 AVI 文件。

1. Alpha 轨道编号：请保证将 00 作为 alpha 轨道编号。目前，不支持其他轨道编号。

示例：将一个 AVI 文件 *smpvid.avi* 中的一个 alpha 频道分配给轨道 00。

```
> medianoche -video00=smpvid.avi -audio00=smpaud.wav -out=sample.usm \
    -alpha00=smpvid.avi
```

audio: 规定音频文件

```
[Format]      -audio##=filename
[Input]       filename : audio file
              ##        : audio track number (## = 00 - 31)
[Function]   Specifies input audio file.
```

1. 音频文件：有关音频文件的详细信息，请参阅第 2.2 部分内容。

2. 音频轨道编号：

- 此处规定的轨道编号将作为音频轨道编号用于 CRI 视频文件。
- 当仅使用一个音频文件时，使用轨道编号 00。
- 最多可规定 32 个音频文件。

示例：将音乐文件、英语语音文件和日语语音文件分别分配给轨道 00、01 和 02。

```
> medianoche -video00=smpvid.avi -out=sample.usm \
    -audio00=music.wav -audio01=voice_en.wav -audio02=voice_ja.wav
```

mca: 为环绕影片规定音频文件

```
[Format]      -mca**_##=filename
[Input]       filename : audio files
              **        : audio track number
              ##        : audio channel number (## = 00 - 05)
[Function]   Specifies audio file for surround movie
```

1. 音频文件：

- 为了创建环绕影片，需要确定相应音频素材文件的扬声器位置。
- 所有环绕影片的素材文件都应当是单声道数据。

2. 频道编号：

- 扬声器位置不容缺省。请保证为所有位置规定素材文件。
- 下表说明了环绕影片在频道编号与扬声器位置之间的映射。

频道编号	扬声器位置
00	左
01	右
02	左环绕
03	右环绕
04	中心
05	低频音效

示例：将环绕影片的音频文件分配给轨道 0.

```
> medianoche -video00=smpvid.avi -out=sample.usm \
-mca00_00=f1.wav -mca00_01=fr.wav -mca00_02=sl.wav -mca00_03=sr.wav \
-mca00_04=c.wav -mca00_05=lfe.wav
```

subtitle:规定字幕信息文件

```
[Format]      -subtitle##=filename
[Input]       filename : subtitle information file
              ##        : subtitle channel (## = 00 - 15)
[Function]   Specifies subtitle information file.
```

1. 字幕信息文件：有关字幕信息文件的详细信息，请参阅第 2.3 部分内容。
2. 字幕频道编号：最多可规定 16 个字幕信息文件。

示例：将英语和日语字幕信息文件分别规定为频道 00 和频道 01.

```
> medianoche -video00=smpvid.avi -audio00=smpaud.wav -out=sample.usm \
-subtitle00=subttl_en.txt -subtitle01=subttl_ja.txt
```

cuepoint: 规定提示点信息文件

```
[Format]      -cuepoint=filename
[Input]       filename: cuepoint information file
[Function]   Specifies cuepoint information file.
```

有关提示点信息文件的更多信息，请参阅第 2.4 部分内容。

示例：

```
> medianoche -in=sample.avi -out=sample.usm -cuepoint=sample_cuepoint.txt
```

3.2.2 视频编辑子命令

crop: 对原图像进行部分剪切

```
[Format]      crop=crx,cry,crw,crh
[Input]       crx: X coordinate of cropped image against the original image
              cry: Y coordinate of cropped image against the original image
              crw: Image width after cropping
              crh: Image height after cropping
[Function]   Crops a part of original image.
只要剪切的部分位于原图像内部，任何数值都是允许的。
默认为无剪切。
```

示例：以 (0, 0) 为基点，剪切的宽度为 640，高度为 480.

```
> medianoche -in=sample.avi -out=sample.usm -crop=0,0,640,480
```

scale: 对原图像进行放大或缩小

```
[Format]      scale=scw,sch
[Input]       scw : the width of resized image
              sch : the height of resized image
[Function]   Scales up/down the original image by averaging pixel value
默认为无放缩。
```

示例：将视频的尺寸重新设置为宽 640，高 480.

```
> medianoche -in=sample.avi -out=sample.usm -scale=640,480
```

3.2.3 Video 编码设置子命令

bitrate: 比特率

```
[Format]      -bitrate=rate
[Input]       rate: bitrate [bps]
[Function]   Specifies the bitrate of output movie file in bps (bits per second).
```

当没有对这一子命令进行规定时，高清晰视频文件在输出时不会重新编码。

默认： 3000000bps.

示例：将比特率设置为 6Mbps.

```
> medianoche -video00=smpvid.avi -audio00=smpaud.wav -out=sample.usm \
    -bitrate=6000000
```

br_range: 可变比特率控制的比特率范围

```
[Format]      -br_range=min,max
[Input]      min: minimum bitrate [bps]
            max: maximum bitrate [bps]
[Function]   Sets the bitrate range with minimum and maximum values for variable
            bitrate control.
```

默认：最小=0bps, 最大=3000000bps

示例：将比特率的范围设置为最小=0bps, 最大=3000000bps.

```
> medianoche -video00=smpvid.avi -audio00=smpaud.wav -out=sample.usm \
    -br_range=0,3000000
```

framerate: 帧速率

```
[Format]      -framerate=decfr
            -framerate=fr_num,fr_den
[Input]      decfr: frame rate in decimal number
            fr_num: numerator of frame rate in fractional number
            fr_den: denominator of frame rate in fractional number
[Function]   Specifies movie frame rate (fps).
```

可使用下列两种格式：

1. 以十进制形式表示： -帧速率=29.97
2. 以分数表示： -帧速率=30000/1001

最高可规定 60 fps。当缺省时，将使用视频素材的原始帧速率。

示例：将视频帧速率设置为 29.97 fps。

```
> medianoche -video00=smpvid.avi -audio00=smpaud.wav -out=sample.usm \
    -framerate=29.97
```

hcfilter: 高压缩过滤器

```
[Format]      -hcfilter=fstype
[Input]      fstype: high compression filter type
            ON: always on
```

```
OFF: always off
AUTO: automatically applies when it's applicable. (Default)
[Function] Applies high compression filter for video encoding.
```

高压缩过滤器在对较低的比特率进行编码时尤为有效。

3.2.4 GOP (图片组)子命令

GOP 是编码视频流中的一组连续图片。每一个编码视频流都包含连续的 GOP。可从它包含的图片中生成可见帧。

GOP 包括下列图片类型：

- I-图片或 I-帧 (内部编码图片) – 参考图片，它表示一个固定图片，并且独立于其他图片类型。每一个 GOP 都以这种类型的图片开始。
- P-图片或 P-帧 (预测编码图片) – 包括动作 – 在处理 I- 或 P-帧时的差异信息进行补偿。
- B-图片或 B-帧 (双向预测编码图片) – 包括 GOP 中之前和之后的 I- 或 P-帧的差异信息。

gop_i: I-图片在 GOP 中的数量

```
[Format] -gop_i=count
[Input] count: number of I pictures in GOP
[Function] Specifies number of I pictures in GOP
```

通过调节 GOP 长度和每一图片类型 (I,P,B) 在 GOP 中的数量，可以提高视频的质量。

- I 图片使用的越多，解码负担就越小。
但 I 图片的数据量比其他图片类型的要大。
- 默认： 1

gop_p: P-图片在 GOP 中的数量

```
[Format] -gop_p=count
[Input] count: number of P pictures in one GOP
[Function] Specifies number of P pictures in one GOP
```

通过调节 GOP 长度和每一图片类型 (I,P,B) 在 GOP 中的数量，可以提高视频的质量。

- P 图片的数据大小介于 I 图片和 B 图片之间。

- 默认: 4

gop_b: B-图片在 GOP 中的数量

```
[Format]      -gop_b=count
[Input]      count: number of B pictures in one GOP
[Function]   Specifies number of B pictures in one GOP
```

通过调节 GOP 长度和每一图片类型 (I,P,B) 在 GOP 中的数量，可以提高视频的质量。

- The more B 图片使用的越多，视频的压缩率就越高，因为在同一比特率中会分配更多的视频信息。但是解码负担会更大。
- 默认: 2

gop_closed: 关闭 GOP

```
[Format]      -gop_closed=sw
[Input]      sw: switch for the closed GOP
              ON: closed GOP (Default)
              OFF: non closed GOP
[Function]   Enables the closed GOP.
```

当启用关闭 GOP 时，GOP 顺序将独立于其他的 GOP 顺序，并且能够方便地进行编辑。

detectsc: 场景变化检测

```
[Format]      -detectsc=sw
[Input]      sw: switch for the scene change detection
              ON : (Default)
              OFF:
[Function]   Enables the scene change detection.
```

在对剪切场景进行检测时，将会自动插入一个 I-图片。

通过在新场景的开始插入一个 I-图片，在场景变化后，该图片质量将会立即得到提高。

3.2.5 音频编码设置子命令

hca: HCA 编解码器的音频源进行编码

```
[Format]      -hca=sw
[Input]      sw: switch for HCA encoding
              ON: Encode as HCA
```

OFF: Not encode as HCA (Default)
[Function] Specifies whether or not to encode audio source as HCA codec.

将此选项设为禁用时，用 ADX 编解码器对音频源进行编码。

hca_quality: 作为 HCA 编码时的声音质量

[Format] -hca_quality=count
[Input] count: Selects encoding quality from the following 5 values
5 = Highest
4 = High (default)
3 = Medium
2 = Low (high compression)
1 = Lowest (highest compression)
[Function] Set sound quality for HCA encoding.

下面显示的是编码质量值与压缩率之间的粗略关系：

质量	单声道	立体声或更多声道
5	1/4	1/4
4	1/6	1/6
3	1/8	1/8
2	1/10	1/12
1	1/12	1/16

3.2.6 杂项

-ms_fullpel_still: 全像素静态部分的动作查询

[Format] -ms_fullpel_still=sw
[Input] sw: switch for the motion search in full pel for still parts
ON:
OFF: (Default)
[Function] Specifies whether or not to perform motion search in full pel for still parts.

当启用这一选项设置时，动作查询将对移动部分执行半像素（半像素距离），而对静态部分执行全像素。这一选项将对黑暗场景的波动作出限制，但无法复制微妙动作。.

cleanup: 删除中间文件

```
[Format]      -cleanup=sw
[Input]      sw = switch for deleting intermediate files
            ON: Delete (Default)
            OFF: Not delete
[Function]   Specifies whether to delete intermediate files that are created during
            encoding process.
```

当设置为禁用时， CRI 影片视频文件 (.usv) 和 CRI 影片音频文件 (.usa) 将会保留在 CRI 影片文件 (.usm) 的同一文件夹中。

默认设置为开启（删除）。

work_dir: 设置工作目录

```
[Format]      -work_dir=dirname
[Input]      dirname : working directory
[Function]   Sets the working directory to create intermediate files for encoding
            process.
```

如果没有规定工作目录，中间文件将会在输入文件的同一目录下创建。

debug: 输出调试信息

```
[Format]      -debug=sw
[Input]      sw = switch for outputting debug information
            ON: Display debug info.
            OFF: No debug info. (Default)
[Function]   Enables to output debug information.
```

默认设置为禁用（无调试信息）。

preview: 显示预览屏幕

```
[Format]      -preview=sw
[Input]      sw = switch for displaying preview screen
            ON: Display (Default)
            OFF: Not display
[Function]   Specifies whether or not to display the preview screen that appears
            during encoding process.
```

默认设置为开启（显示）。