

VCB-Studio 教程 03: 基础工具的安装和调试

本教程讲述如何安装并调试 avisynth+, avspmod, VapourSynth, VapourSynth Editor, MeGUI, mkvtoolnix 和 MediaInfo。调试的方法中顺道介绍常见的源-成品对比手段。

1. avisynth+ 和 avspmod 的安装

avisynth 是一直以来滤镜处理的传统工具，后来分裂成 avisynth 2.6.0(简称 avs 2.6) 和 avisynth+(简称 avs+)，两者大同小异。本系列教程如无例外，均以 avs+ 32bit 为准。

avs+只允许标准安装，不像 avs 2.6 允许标准安装和绿色安装，容易造成版本和滤镜目录的混乱。

SVP(Smooth Video Project)依赖 avs 运行。所以如果你需要使用 svp，不建议你重新安装 avs+。

安装 avisynth+之前，先去控制面板-程序，找到之前所有的 avs 版本，卸载。

如果卸载完了有弹窗问你是否要保留滤镜目录，选否。

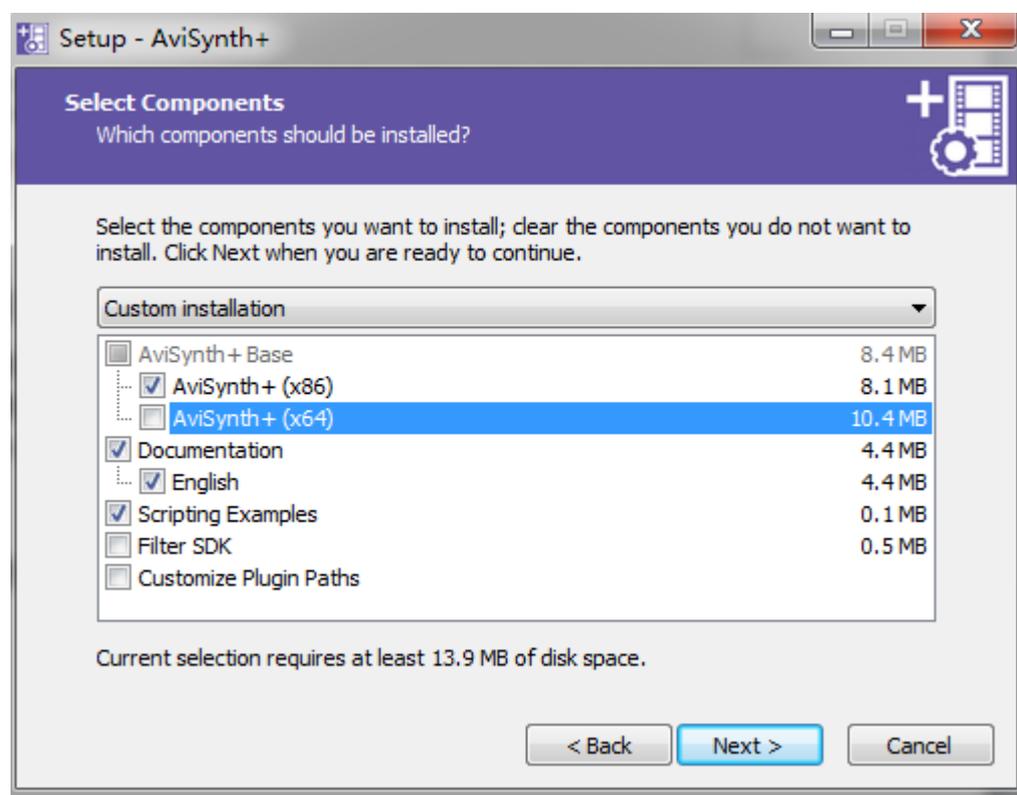
然后安装 avisynth plus 的本体：

<https://github.com/AviSynth/AviSynthPlus/releases/download/Rel-r1576/AviSynthPlus-r1576.exe>

Language 选 English。

安装目录，千万记好你装到哪里。

随后的安装选项，你可以根据你的需要选择 32bit(x86)还是 64bit(x64)，一般情况下推荐只安装稳妥的 32bit，毕竟 avisynth 官方没有 64bit 版本，一些组合滤镜也很难把所有插件找齐 64bit 版本：



32bit 的 avs，一般被 32bit 的自行环境启动，比如说 32bit 的播放器，32bit 的 x264。如果要想让 64bit 的编码器读取 32bit 的 avs，则要借助 avs4x264.exe 这样的 pipe 工具。

同理，64bit 的 avs 只能被 64bit 的环境自动启动。如果你装了 32bit 的 avs，而你的播放器是 64bit 环境，那么播放器是无法打开 avs 的。

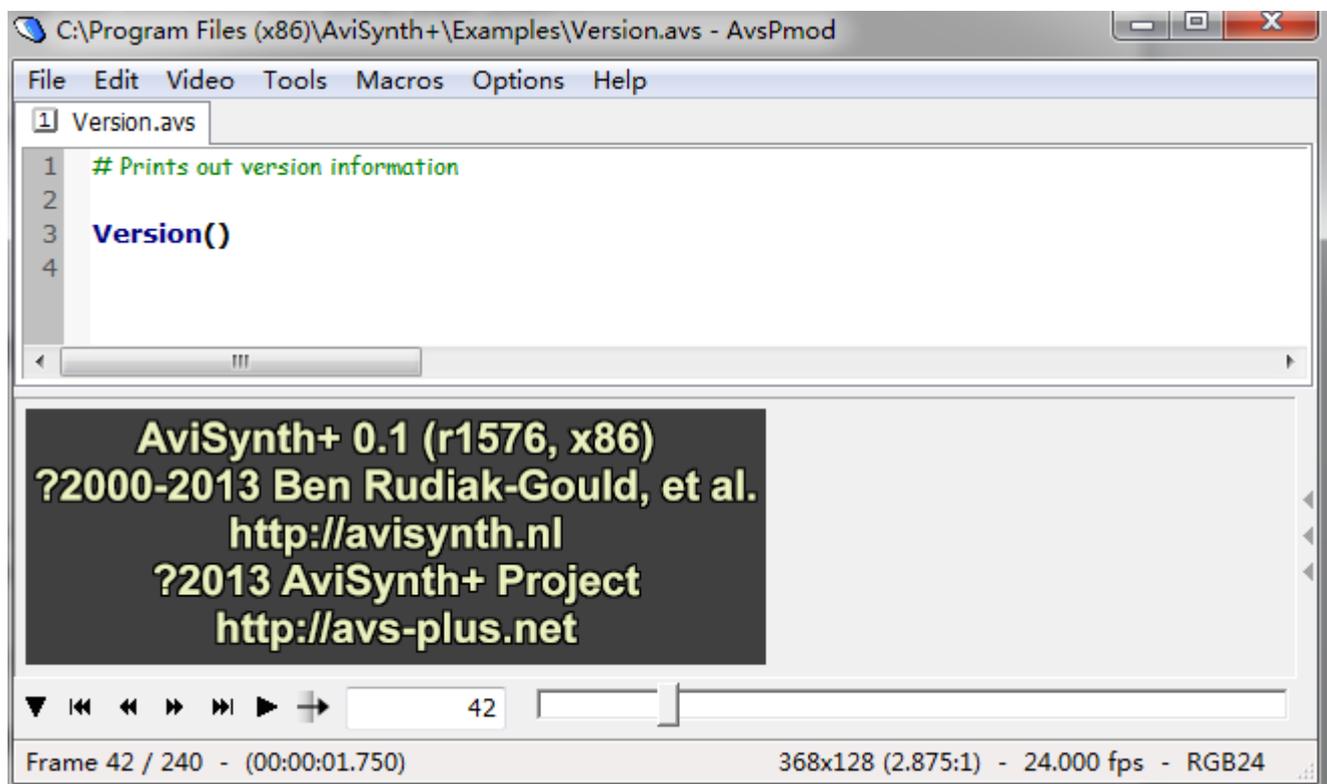
avs 脚本是一种编程语言。一般后缀就用 .avs，通过安装的 avisynth 环境（最核心的是 avisynth.dll，一般非绿色安装的时候会丢 system32/syswow64）来将脚本转化为视频和音频，以模拟一个 avi 文件。

最简单的 avs 脚本大概算是 AviSynth+\Examples 目录下的 Version.avs:

Version()

这个函数的作用就是输出当前安装 avs 的版本。

调试 avs 最常用的工具为 avspmod。本教程中已经附带绿色版。下载解压到电脑中，双击 avspmod.exe，就可以使用了。建议把它设置为 avs 文件的打开方式，比如我们用它打开 version.avs，拖动一下下方的进度条以显示内容：

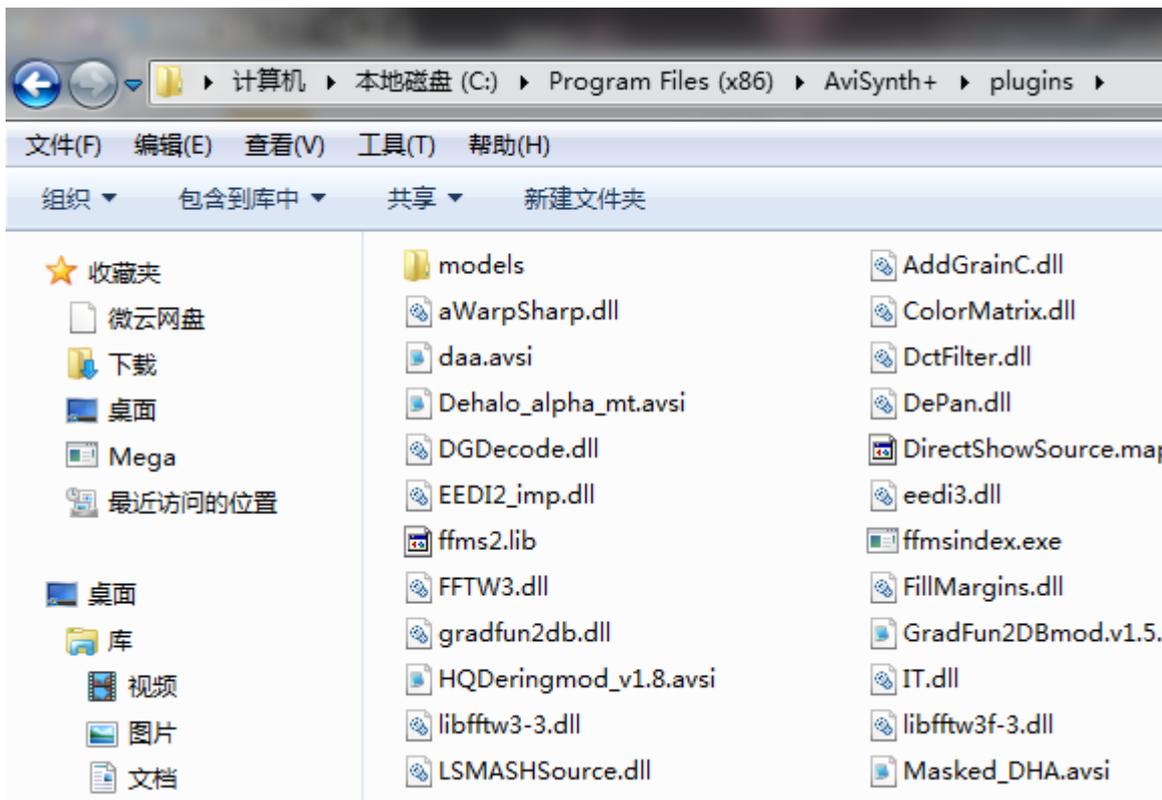


如图，avspmod 就可以显示 avs 文件想要表达的内容，这里显示的是版本。

avs 的运行依赖大量插件，一般推荐把插件放在自动加载的目录中。avs+ 的自动加载目录为根目录下的 plugins（64bit 则为 plugins64）。你也可以放在 plugins+/plugins64+ 中，都没有关系。

附件中的 plugins.7z 是我个人积累的滤镜，**不保证最新**。mawen1250 也有发过自己的滤镜包，需要的自己去 nmm 找。下载后，解压到 plugins 文件夹中就好。

解压覆盖后的效果大概如下图：

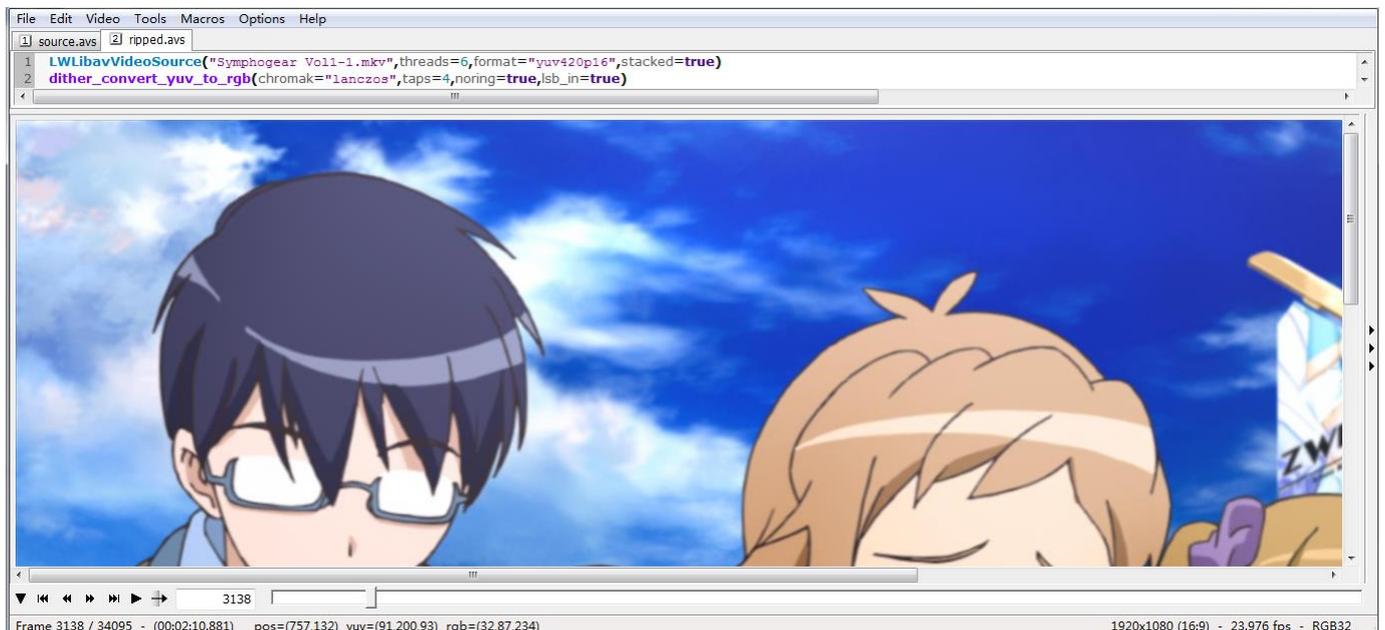


检验安装成果可以用这个简单的脚本：

```
LWLibavVideoSource("C:\xxxxx.mkv",format="yuv420p16",stacked=true)
dither_convert_yuv_to_rgb(chromak="lanczos",taps=4,noring=true,lsb_in=true)
```

C:\xxxxx.mkv 是一个输入的视频文件，也可以是 MP4/m2ts 等。如果 avs 文件和视频文件在同一个目录，还可以省略绝对路径，直接用文件名。第一行的意思是把它以 yuv420 16bit 精度读入，第二行的作用是高质量转为 RGB24。用 avspmod 预览，就应该可以模拟播放器播放视频的画面。

avspmod 可以同时载入多个 avs。如果它们输出图像的时间长度，分辨率都一样，切换标签的时候，会自动同步到同一帧数，和同一画面位置。这点很适合用来比较多个视频，特别是用来比较源、成品和多版本 Rip：



2. VapourSynth 和 vsedit 的安装

VapourSynth (简称 VS) 是新时代的处理工具, 先进性表现为依赖 Python 的强大可编程性, 原生的多线程优化, 大幅提升的内存管理, 以及官方 64bit 版本。缺点就在于 vs 实在是太新了, 教程实在是少的可怜, 包括本系列教程中绝大部分, 初稿均写在 vs 普及之前。

好在熟悉 avs 的人, 适应 vs 并没有太多难度, 而力图讲述处理原理和思路的教程, 基于 avs/基于 vs 也不重要。

安装 VS 之前, 一般需要先安装 Python 64bit:

<https://www.python.org/ftp/python/3.5.1/python-3.5.1-amd64.exe>

老规矩, 记好你安装到了哪里; 因为之后需要往里面丢东西。

VS 本体的最新版可以在这里下载: <https://github.com/vapoursynth/vapoursynth/releases>, 一般选择最新的稳定版。安装的时候只选择稳定版, 同样, 记好你安装到了哪里。

VS 的插件分两类, 一类是编译好的 dll, 一类是写好的 library。前者默认放置位置是 VapourSynth\plugins64, 后者是 Python35\Lib\site-packages 文件夹。就我收集范围内的, 前者在 VS plugins64.7z 中, 把各种 dll 啥的解压到 plugins64 文件夹内。后者 (3 个 py 文件) 在 VS libraries.7z 中, 解压到 Python 安装目录里 Python35\Lib\site-packages 内就好。

安装好 vs 插件之后, 我们需要安装 vs 的调试工具, vsedit:

https://bitbucket.org/mystery_keeper/vapoursynth-editor/downloads

一般选择最新的 64bit 版本下载。

vs 脚本一般以 vpy 作为结尾。这里我们可以用这个简单的脚本来测试效果:

```
import vapoursynth as vs
import sys
import havsfunc as haf
import mvfunc as mvf

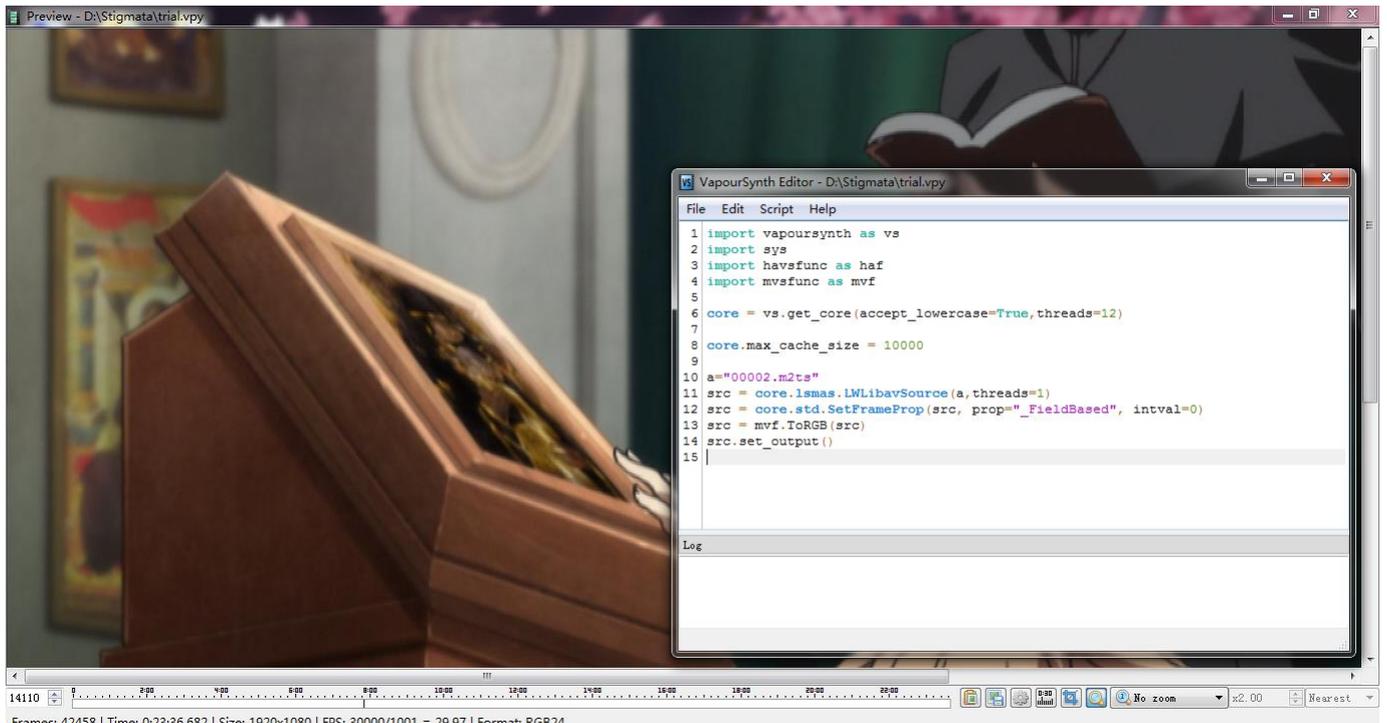
core = vs.get_core(accept_lowercase=True, threads=8)

core.max_cache_size = 2000

a="00002.m2ts"
src = core.lsmas.LWLibavSource(a, threads=1)
src = core.std.SetFrameProp(src, prop="_FieldBased", intval=0)
src = mvf.ToRGB(src)
src.set_output()
```

其中, a="00002.m2ts" 一句可以换成任何视频都行; 如果 vpy 文件和视频不在同一个文件夹, 则需要用绝对路径。这个 vpy 脚本先是载入相关的库到 Python 环境, 然后设置线程和缓存, 接着就是本体。先通过 a 来指定文件名, 然后用 LAV 读入视频, 指定为 progressive 格式, 接着用 mvf 库中的函数转为 RGB24, 最后输出。

点击 Script-Preview(F5), 可以预览效果:



如果出错，在 log 标签卡中会显示出错信息。

如何用 VapourSynth 实现两个视频同时预览，一般用这种方法：

```

import vapoursynth as vs
import sys
import havsfunc as haf
import mvsvfunc as mvf

```

```

core = vs.get_core(accept_lowercase=True,threads=8)
core.max_cache_size = 2000

```

```

source = "00001.m2ts"
ripped = "Symphogear Vol1-1.mkv"
src16 = core.lsmas.LWLibavSource(source,format="yuv420p16")
rip16 = core.lsmas.LWLibavSource(ripped,format="yuv420p16")

```

```

res = core.std.Interleave([src16,rip16])
res = mvf.ToRGB(res,full=False,depth=8)

```

```

res.set_output()

```

就是同时载入两个视频，然后前后间隔显示每一帧，再转为 RGB。预览的时候，0,2,4,6,8...帧是源，1,3,5,7,9...帧是成品。如需放大，可以在右下方设置 zoom，放大倍数自己写，放大方法选 Nearest:



3. MeGUI 的安装

MeGUI 是各类压制软件 GUI 中，功能最齐全的。即便不用它做日常压制，它提供丰富的组件下载和更新，也让它成为一个不可或缺的工具：

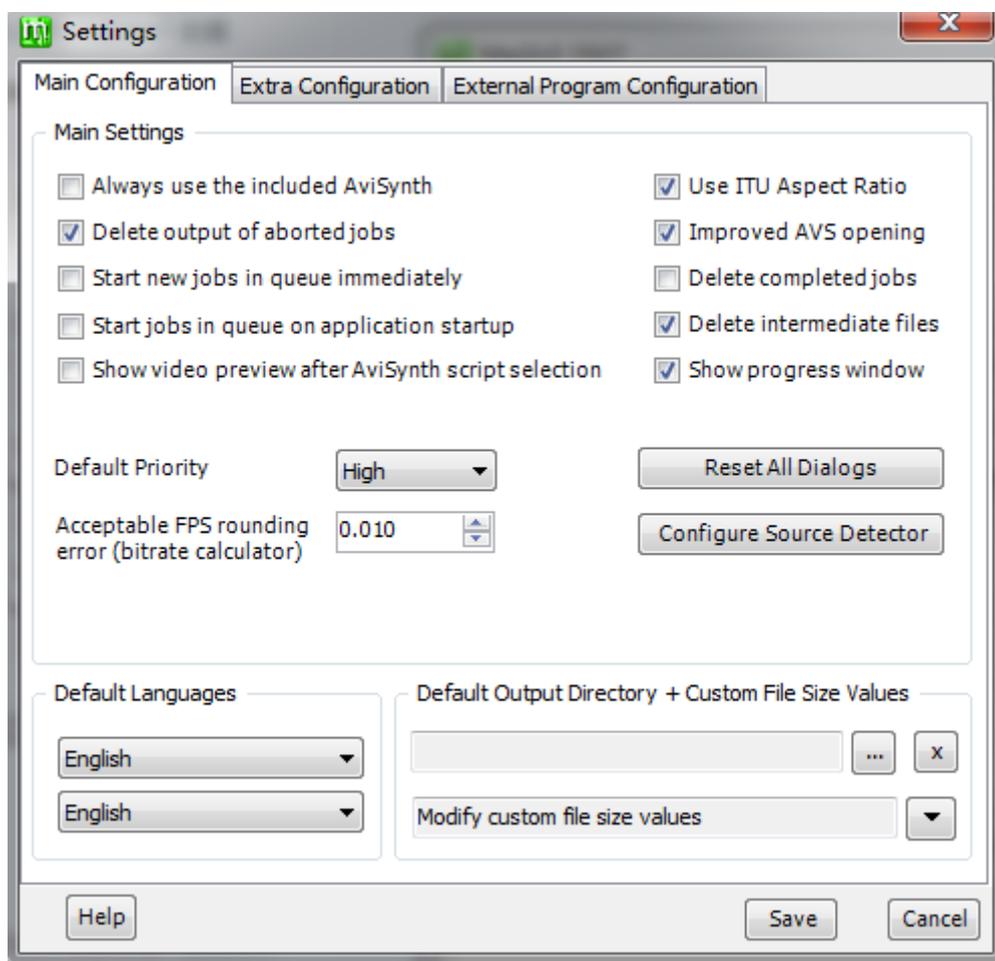
<http://sourceforge.net/projects/megui/files/latest/download>

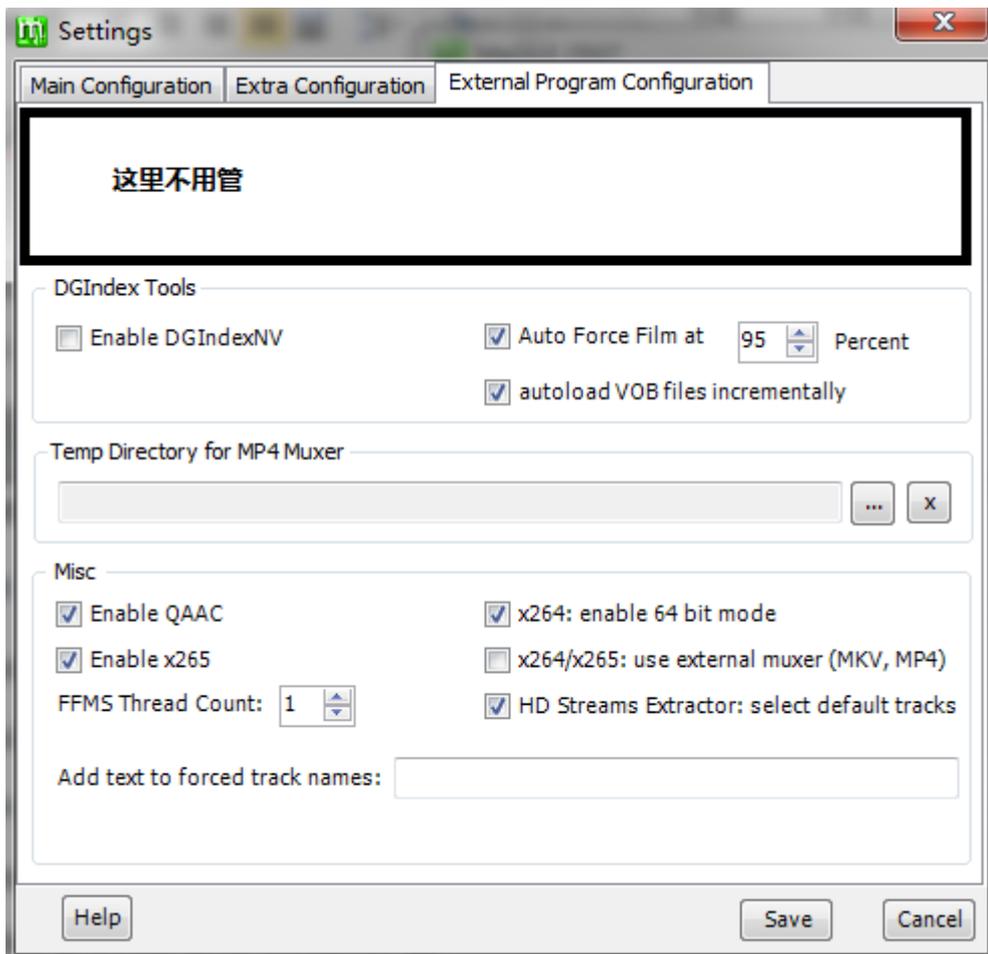
安装之后，如果有需要让你更新的，全部更新
MEGUI 的根目录力求简单，特别是不要有中文字符

装好之后，右键 megui 根目录-属性-安全-编辑，允许一般用户对 megui 拥有全部读写权限。

MeGUI 主界面：Options-settings

按照如下勾选：





勾选好之后，MeGUI 会提示更新，加入 qaac 和 x265 两个组件。

出于授权因素，MeGUI 需要自己安装 QAAC 的必备组件（苹果的 Quicktime）。

去 http://www.free-codecs.com/qt_lite_download.htm 安装 QuickTime Lite，如果机器上有 Quicktime/itunes（可能）无需安装。

x265 的作用无需多解释；QAAC 是 alac 无损格式的编码器，同时，它编码的 Apple AAC 被广泛认为是最优秀的 AAC 编码方案之一。

4. MKVToolnix 的安装

MKV 是我们日常压制见到的最多的格式，为了方便对 mkv 的处理，我们还需要安装一下 mkvtoolnix，这个工具将方便我们以后对 mkv 格式的文件进行操作。下载地址：<http://www.fosshub.com/MKVToolNix.html> MeGUI 中自带了简化版的 mkvtoolnix（或者说 mkvmerge.exe，负责合成 mkv 的执行文件，隶属于 mkvtoolnix 的一部分），在 MeGUI\tools\mkvmerge 中。我们只需要从网站上下载 64bit 的 portable，解压覆盖到 mkvmerge 文件夹中就好了。

另一个实用的小工具是 MKVExtractGUI2。从名称可知，它是负责拆分 mkv 文件的 mkvextract.exe 的 GUI。下载地址：<http://sourceforge.net/projects/mkvextractgui-2/> 下载后，把 exe 解压到 mkvmerge 的文件夹中就可以了。

MKVtoolnix 更新频繁，一般见到要你更新，可更可不更。只不过更新之后用它封装，一定要仔细检查下播放效果——你不知道这玩意更新了引入什么 bug

5. MediaInfo 的安装和使用

mediainfo 是最常用的查看媒体文件信息的工具。本体可以在这里下载：

<https://mediarea.net/zh-CN/MediaInfo/Download/Windows>

mediainfo 显示一个文件的主界面是这样：



在这个主界面中，很容易看到分辨率、帧率、bitdepth 等信息（High/Main 是 8bit，High 10/Main 10 不出意外是 10bit），可以看到音轨的数量（主界面最多显示两条）

点击左边栏倒数第二个图标，文本，则可以看到更详细的信息，包括编码参数。所有的轨道，以及章节。

mediainfo 一般用来检查一些播放时候注意不到的问题，比如 bitdepth，编码参数，音轨章节信息等。