

Quick Cut ——你最得力的视频处理工具 正式发布

作者: [HaujetZhao](#)

原文链接: <https://ld246.com/article/1595657861530>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

Quick Cut

Quick Cut 是一款轻量、强大、好用的视频处理软件。它是一个轻量的工具，而不是像 Davinci Resolve、Adobe Premiere 那样专业的、复杂的庞然大物。Quick Cut 可以满足普通人一般的视频处理需求：压缩视频、转码视频、倒放视频、合并片段、根据字幕裁切片段、自动字幕、自动剪辑.....

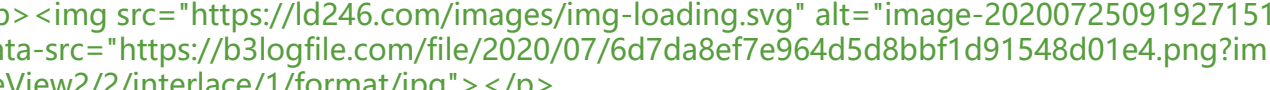
它是开源的，你可以免费使用它。

开源地址： <https://ld246.com/forward?goto=https%3A%2F%2Fgitee.com%2Fhaujet%2FQuickCut> <https://gitee.com/haujet/QuickCut>

发行版下载地址： <https://ld246.com/forward?goto=https%3A%2F%2Fgitee.com%2Fhaujet%2FQuickCut%2Freleases> <https://gitee.com/haujet/QuickCut/releases>

下载后，无需安装，解压即用。

界面预览：



--背景

好几年前，有一次想将一个视频中的片段剪出来，才发现，市面上根本没有给普通用户用的视频处理软件。

我去百度、知乎上搜【视频剪辑软件】，陆续得到了以下结果：

Adobe Premiere， 正版一年上千元、随时【停止运行】的专业剪辑软件

Vegas 等专业软件就不再列了

爱剪辑， 一个导出就带推广片头的中文剪辑软件

格式工厂， 一个老牌的国产转码编辑软件

小丸工具箱， 一个从 Bilibili 知道的压制软件

还有大大小小的其它软件

我就只想剪一小个片段啊！专业软件学不起、买不起、电脑太卡带不起！可是其它小软件、许多剪辑软件，都有这一些问题：

国外软件选项太多，各种参数看不懂。

多数免费软件都带水印，或者导出后画质会下降，要么导出的体积巨大，耗时长不说，还不是无剪辑。

用的最好的还是 格式工厂 和 小丸工具箱。但他们功能很少，还有小丸工具箱，官网的下载地址是百度网盘链接已经挂了，也不知道从第三方下载的版有没有木马、广告.....

后来，从视频压制这个话题，我知道了 FFmpeg 这个神级开源工具，它的能之强大、应用之广泛到惊了我的下巴！但它是个命令行工具，到百度上一搜，也只是些文章说怎样的命令可以做什么、常用命令有什么。相关的图形界面工具，一个好用的都没有！（当然要点名表下 Lossless Cut，它还是很好用的，只是功能太少）。没有一个软件能让 FFmpeg 在普通大众用户中发挥潜力，真是可惜了！

于是一通操作，把 FFmpeg 的官方文档逐句做了翻译，记了笔记，然后用它来处理视频，真是快！免费、无广告，剪出来的视频也可以画质无损。100 兆的视频压缩到 10 兆，肉眼画质无损，那一个巴适！

但是有一个问题！每次要处理一个视频，都要手动输入命令行，非常的麻烦，有时候忘记执行某操作是用哪个参数了，还要去翻笔记，难受！于是我就想做一个图形界面工具，想要做什么，在里面两下，命令参数啥的都是自动生成，最后点击运行就好。于是先后做了基于 Tasker 的安卓端 FFmpeg GUI Tool、基于 Quicker 的 FFmpeg GUI Tool。

但是上面两个平台都局限太多，想要好用，还是得自己做 GUI。我自己只是入门 python 的水平只能硬着头皮花了几天在学习平台 B 站看完了 PyQt 的入门。然后边查边做，因为 PyQt 的注释很少

有时一个简单的小细节要花半天到一天处理。最后，成品还是出来了！

当然，除了 **FFmpeg** 的功能外，我还做了 **语音自动转字幕** 功能。这里，我要不指名地 Diss 一些商家：

- 首先是价格，贵的要 1 元/分钟，便宜些的也要 4 毛/分钟！转个 20 分钟的视频，就要收我 8 元抢钱呐！啊？你们是没有语音引擎的，都是用的 API，大批量采购，你们用的 API 价格也就 1.2 元/小时左右吧！用 API 成本几十倍的价格提供服务，这吃相，真是难看。
- 然后是上传视频。语音转字幕，上传音频便足以，一些商家却需要上传完整视频！2GB 的视频，常其音频只有几十 MB 左右。你们是你们觉得：用户见转写了几十 MB 的视频，会觉得这么小的文，1 元/分钟的价格不值，转写了 2GB 的视频，一看这转写体积，就觉得花钱花的值？

在 Quick Cut 里，我做了 **语音自动转字幕** 功能，可以使用阿里或腾讯的 AI。就以阿里的 API 为例，普通元套餐的商用价格是 2.5 元/小时，四十五几乎就是不要钱！下面的帮里，我写下了申请阿里 API 的教程，只要填入 API，就可以使用语音转字幕了。

另外，我还将另一个 **自动剪辑神器** 放进了 Quick Cut，将你的 Vlog、视频教程一键自动剪好。下面会有详细介绍。

同时，推荐 Quick Cut 中一个对学习外语的同学非常有用的功能：**将字幕中的每一句对应的视频剪成片段提取出来**。用于制作外语学习的视频素材爽的不要不要的！应该算是 **Anki** 用户的一个福利功能吧！

--特性>:sparkles: 特性

- 简单的界面
- FFmpeg 预设丰富
- 可自定义预设
- 合并视频片段
- 逐句提取每句字幕对应的视频片段
- 自动转字幕
- 自动剪辑
-更多待探索

--界面和功能介绍>:crystal_ball: 界面和功能介绍

FFmpeg-界面>FFmpeg 界面

在这个界面，你可以使用许多 ffmpeg 预设，对音视频进行处理，比如说：

我想将手机上录制的视频压制下，减小它的大小，那么只需要：先在 **输入 1** 框输入待压制的视频文件（此时会自动生成输出文件名），再在右侧预设列表选择 **H264 制** 预设（此时会自动生成总命令），最后点击底部的 **运行** 按钮，会启动压制了。

[点击去查看 Quick Cut 压的视频教程](https://www.bilibili.com/video/BV18T4y1E7FF?p=1)

[点击去查看 Quick Cut 预讲解视频教程](https://www.bilibili.com/video/BV18T4y1E7FF?p=2)

分割视频界面>分割视频界面

这个界面有三个功能：根据字幕分割视频、根据大小分割视频、根据时长分割视频。

根据字幕分割视频是个神级功能，尤其适合于制作外语学习的视频素材。将例如美剧的视频放进，再把相应的字幕文件放进去，就可以将每一句字幕对应的视频片段剪出来！如果你的字幕时间轴和频时间轴有偏差，还可以进行手动的校准。

根据时间分割视频和根据大小分割视频主要是针对分享到短视频平台和微信平台的。

[点击去查看 Quick Cut 分视频的视频教程](https://www.bilibili.com/video/BV18T4y1E7FF?p=3)

 data-src="https://b3logfile.com/file/2020/07/8096d34c3e70482890f8545a21887d06.png?imageView2/2/interlace/1/format/jpg"/>

合并片段界面>合并片段界面

这个简单，要合并的视频拖进去，调整下顺序，点击运行，就可以将这些视频合并成一个文件。

ig 下载的 15 秒 story 视频片段就可以轻松合并啦! </p>

<p>点击查看 Quick Cut 合片段的视频教程</p>

<p>#</p>

<h2 id="下载视频界面">下载视频界面</h2>

<p>这个界面提供了两个命令行工具的图形界面用于下载视频，最简单的用法就是将链接复制进去，后点击下载。支持的网站有很多比如优酷、B 站、YouTube、P 站（逃）.....</p>

<p>另外你还可以在里面设置 cookies，就能够用你大会员身份登录的 cookie 信息下载大会员视频质了。</p>

<p>点击查看 Quick Cut 下载的视频教程</p>

<p>#</p>

<h2 id="自动剪辑界面">自动剪辑界面</h2>

<p>自动剪辑的原理是通过给视频中有声音的片段和没有声音的片段施加不同的播放速度，达到只保留关键信息部分的效果，非常适合做 vlog 和视频教程。</p>

<p>同时你也可以选择使用阿里云或者腾讯云的语音服务，先将视频转出字幕之后，再根据字幕中的关键词对视频片段进行保留和删除操作。</p>

<p>除了下面那个演示视频，在本教程中的所有视频，都使用了自动剪辑，然后才上传的。</p>

<p>点击查看 Quick Cut 自动剪辑的效果演示视频</p>

<p>点击查看 Quick Cut 自动剪辑的视频教程</p>

<p></p>

<h3 id="自动转字幕界面">自动转字幕界面</h3>

<p>只要将你的视频或者音频文件拖进去，然后点击运行，就可以生成一个 srt 格式的字幕。</p>

<p>语音识别方面使用了阿里云或者腾讯云的引擎，准确率有 95% 以上。如果想给自己的视频配字，就可以先用这个功能，自动转出字幕之后，再手动修改一下里边偶尔的错别字，效率非常高。</p>

<p>语音识别引擎需要用户自己去阿里云官网申请 API 才能用（对申请过程我做了视频教程）。阿里云的语音服务开通后，每个用户有 3 个月的免费试用时间，在这 3 个月内，每天都有两小时的录音文件转换额度。试用期过后，业版的价格是每小时音频转换 2.5 元，随着使用量的增加，这个价格还会更低。如果买 1000 小时的餐，价格可以低到 1.5 元每小时。</p>

<p>如果你对比一下目前网上能找到的视频转字幕服务，你就会知道 Quick Cut 的这个转字幕功能多便宜：</p>

网易见外工作台，普通的视频转字幕，收费 1 元每分钟，60 元每小时。英文转字幕价格翻倍。

号称全网最低价的突字幕，收费 0.005 元每秒，3 毛每分钟，18 元每小时。

Arctime 收费 30 积分每分钟，也就是 3 毛每分钟，18 元每小时。

<p>来对比一下：阿里云 2.5 元每小时，前三个月每天免费用两小时，用得越多价，格还会更低。</p>

<p>腾讯云方面的价格还会更低，只是转换速度没有阿里云快。所以推荐使用阿里云。就算是用商业每小时 2.5 元的价格也不心疼。</p>

<p>如果你急着把最新的美剧视频下载下来，字幕组还没有出字幕，但是生肉太难啃，就可以用这个字幕功能，将英文字幕转出来，配合着看。</p>

<p>如果你是视频工作者、UP 主、视频公司负责人，平常有大量的视频需要转换字幕，用这个就可舍去使用其他网上平台导致的每小时几十元的成本，一个月下来成本或许能节约到上千元。。</p>

<p>哎，说实话，要是这软件推广开来，对一些视频转字幕的服务商，真的是断人财路，杀人父母。</p>
<p>点击查看 Quick Cut 自转字幕的演示</p>
<p>#</p>
<h2 id="语音识别输入界面">语音识别输入界面</h2>
<p>如果你已经添加上阿里云的语音引擎，就可以使用阿里云的一句话识别服务进行语音识别输入。任意界面，只要长按键盘上的盗窃锁定键超过 0.3 秒钟，就可以开始说话，进行语音识别。松开按键后，识别结果会自动打到输入框。</p>
<p>有人推荐使用讯飞的语音识别，但是讯飞语音识别的快捷键在 F6 那个位置，离主键盘的位置很远，手指够过去费劲；而且那里有很多大小一样的按键，这个快捷键也不好找；他还会和很多软件的快捷键冲突。</p>
<p>而长按大写锁定键进行语音识别就非常方便，离打字的手非常近，需要的时候很轻松就能按到，不会有快捷键冲突。</p>
<p>阿里云语音识别的准确率非常高，这篇文章大部分都是用这个语音识别功能打的，然后修改了少错字。</p>
<p>点击查看 Quick Cut 语音识别输入的演示</p>
<p>#</p>
<h2 id="设置界面">设置界面</h2>
<p>在设置界面你就可以在这里配置语音识别的引擎。</p>
<p>勾选上 点击关闭按钮时隐藏到托盘，就可以让软件常驻到后台，方便快捷调出来剪辑视频、语音输入。Quick Cut 的资源占用量非常少，可以放心的放到后台，不影响性能。</p>
<p>点击查看 Quick Cut 配阿里云语音识别引擎的视频教程</p>
<p></p>
<h3 id="帮助界面">帮助界面</h3>
<p>在帮助见面里，有如下按钮，见名知意：</p>
<p></p>
<h2 id="--开发">:hammer: 开发</h2>
<p>这一节普通用户就不用看了。你们只要到发布界面下载使用已打包好的软件就行了，如果你想对代码进行修改，就可以看下这一节。</p>
<h3 id="搭建环境">搭建环境</h3>
<p>你需要 pip 安装这些包：</p>
<pre><code class="highlight-chroma">srtkeyboardnumpysetuptoolsaliyun-python-sdk-corePyQt5audiosmscipycos-python-sdk-v</pre>

tencentcloud-sdk
python
oss2
pyaudio
</code></pre>
<p>其中, pyaudio 很难安装! 编译成功有很多要求。所以我们可以直接到 <a href="https://ld246.com/forward?goto=https%3A%2F%2Fwww.lfd.uci.edu%2F%7Egohlke%2Fpythonlibs%2F%23p
audio" target="_blank" rel="nofollow ugc">这里 下载上它已经被志愿者编译好的 whl 包,
pip 安装, 注意下载对应你 python 版本的包。</p>
<h4 id="阿里云语音识别-sdk">阿里云语音识别 sdk</h4>
<p>然后还需要安装阿里云语音识别引擎的 sdk, 这个 sdk 是无法通过 pip 安装的, 而是要通过 <a href="https://ld246.com/forward?goto=https%3A%2F%2Fhelp.aliyun.com%2Fdocument_detail
2F120693.html%3Fspm%3Da2c4g.11186623.6.569.27675df0FENQ6O" target="_blank" rel="n
follow ugc">这篇阿里云官方文档 的方法进行安装: </p>
<pre><code class="highlight-chroma"><span class="highlight
cl">pip install setuptools
</code></pre>
<p><a href="https://ld246.com/forward?goto=http%3A%2F%2Fdownload.taobaocdn.com%
Ffreedom%2F33762%2Fcompress%2Falibabacloud-nls-python-sdk.zip" target="_blank" rel="n
ofollow ugc">下载 Python SDK。</p>
<h2 id="打包-python-setup-py-bdist-egg---安装-python-setup-py-install">打包 python setup
py bdist_egg # 安装 python setup.py install</h2>
<h3 id="运行问题-">运行问题: </h3>
<p>安装完依赖之后开始运行脚本, 你可能会遇到这些问题: </p>

安装完依赖后, 你运行脚本, 却发现 <code>import oss2</code> 时提示出错: <code>No
odule named 'winrandom'</code>, 这时, 你需要修改 <code>Python38\Lib\site-packages\Cr
pto\Random\OSRNG</code> 下的 <code>nt.py</code> 文件, 将 <code>import winrando
</code> 修改为: <code>from Crypto.Random.OSRNG import winrandom</code>

<p>这些问题的原因是一些模块用了其它依赖, 而这些其它依赖已经好久没更新了。只能遇到一个问
就搜一下, 解决掉。上面这些问题都是我遇到后, 在网上找了解法, 根据自己的情况做了改动, 使得
题得以解决。</p>
<p>搭建好环境可以运行之后, 如果修改了源代码然, 后要进行发布, 就应当打包成可以独立运行的
xe 文件, 下面提供两种方法进行编译打包: </p>
<h3 id="使用-nuitka-编译-">使用 nuitka 编译: </h3>
<p>使用它打包出来的程序运行更快, 只是相对步骤多一些。</p>
<p>先下载安装上 <a href="https://ld246.com/forward?goto=https%3A%2F%2Fsourceforge.n
t%2Fprojects%2Fmingw-w64%2Ffiles%2F" target="_blank" rel="nofollow ugc">MinGW64 8.
</p>
<p>再安装上 nuitka: </p>
<p>pip install nuitka</p>
<p>在实际编译的时候, 先使用这个命令进行编译: </p>
<pre><code class="highlight-chroma"><span class="highlight
cl">start cmd /k nuitka --mingw64 --standalone --show-progress --show-memory --plugin-
nable=qt-plugins --plugin-enable=pylint-warnings --recurse-all --recurse-not-to=numpy,jinj
2 --windows-icon=icon.ico --nofollow-imports --assume-yes-for-downloads --output-dir=out
QuickCut.py
</code></pre>
<p>这时, 可能会遇到如下错误: </p>
<p></p>

<p>就是因为代码里面有中文字符，默认使用的英文编码在转码时出现了错误，你就需要找到并打开后指示的那个 MainControl.py 文件，将里面出错的那一行的 latin1 修改为 utf-8，然后就可以编译成功。</p>

<p>以后其实还并不能运行，因为 nuitka 对于某些包并不会自动导入，还需要你手动导入才行。</p>

<p>所以在实际编译的时候，还是先使用这个命令进行编译：</p>

```
<pre> <code class="highlight-chroma"> <span class="highlight-line"> <span class="highlight-cl">start cmd /k nuitka --mingw64 --standalone --show-progress --show-memory --plugin-
nable=qt-plugins --plugin-enable=pylint-warnings --recurse-all --recurse-not-to=numpy,jinj
2 --windows-icon=icon.ico --nofollow-imports --assume-yes-for-downloads --output-dir=out
QuickCut.py
```

```
</span> </span> </code> </pre>
```

<p>然后再编译的导出目录在命令行窗口中将 QuickCut.exe 打开，会看到类这样的错误：</p>

<p> </p>

<p>这时候，就需要亲自到 Python 的安装目录下找到这个包：</p>

<p> </p>

<p>将它手动复制到导出目录的根目录中：</p>

<p> </p>

<p>重复运行，再次出现包缺失：</p>

<p> </p>

<p>都需要找到这个包：</p>

<p> </p>

<p>粘贴到编译导出目录的根目录中。这个过程要重复很多次，直到将所有缺失的包都复制到导出目。</p>

<p>你可以将这些需要手动导入的包都再复制到另一个文件夹做备份，再次编译后，直接复制进去就。</p>

<p>编译完成，你启动后可能是这个样子的：</p>

<p> 长得很丑对吧。这是因为缺失了 styles 文件夹，你需要到 PyQ5 的安装目录，找到 styles 文件夹：</p>

<p> </p>

<p>将其复制到编译导出目录的根目录之后，再次启动就正常了。</p>

<p>还有，记得将 icon.ico、sponsor.jpg、README.html 复制到编译导出的文件夹，没有图标文的话，就无法启动托盘。</p>

<p>再次编译。测试成功后，就可以使用下面这个命令进行编译：</p>

```
<pre> <code class="highlight-chroma"> <span class="highlight-line"> <span class="highlight-cl">start cmd /k nuitka --mingw64 --windows-disable-console --standalone --show-progress
-show-memory --plugin-enable=qt-plugins --plugin-enable=pylint-warnings --plugin-enabl
=numpy --recurse-all --recurse-not-to=numpy,jinja2 --windows-icon=icon.ico --nofollow-im
orts --assume-yes-for-downloads --output-dir=out QuickCut.py
```

</code></pre>

<p>这样就可以将命令行窗口隐藏。</p>

<h3 id="pyinstaller-编译-">pyinstaller 编译: </h3>

<p>先安装上 pyinstaller : </p>

<p>pip install pyinstaller</p>

<p>直接使用这个命令进行编译: </p>

<p>start cmd /k pyinstaller --hidden-import pkg_resources.py2_warn --noconfirm -w -i icon.ico QuickCut.py</p>

<p>其中, <code>--hide-import pkg_resources.py2_warn</code> 这一句比较重要, 其实整个件并没有用到这个模块, 但是 pyinstaller 他会自动的编译进去, 可是电脑上又没有安装这个模块, 后编译出来的软件就无法运行, 所以一定要加上这个选项。</p>

<h2 id="--打赏">打赏</h2>

<p>万水千山总是情, 一块几块都是情。本软件完全开源, 用爱发电, 如果你愿意, 可以以打赏的方式支持我一下: </p>

<p></p>

<h2 id="--交流">grinning: 交流</h2>

<p>如果有软件方面的反馈可以提交 issues, 或者加入 QQ 群: 114662791</p>

<h2 id="--鸣谢">:pray: 鸣谢</h2>

<p>感谢知乎上的 @Python 与模具 的助, 让我得以用 nuitka 对软件进行打包。</p>