**视频音频相关AI技术调研报告**

1. **调研方式**

根据需求与heygen相关技术的研究与分析，在各大论坛与平台寻找总体解决方案与技术路径，通过阅读大量文献、技术分享以及说明文档一步一步推进项目的调研与实现，并追求各个不同技术之间的配合，实现更大的成果。

**二．一个完整的ai video translation项目是如何实现的**

1. Extract audio from video - ffmpeg
2. Extract text from audio - whisper, paddle speech
3. Translate the text - googletrans
4. Voice synthesis(Text to speech) - tts, paddle speech
5. Voice conversion(Singing voice conversion) - so-vits-svc
6. Lip sync - video-retalking, Wav2Lip
7. separate audio into vocal and instrument - python-audio-separator

https://github.com/pranauv1/AI-Video-Translation/blob/main/Video\_Translation\_%26\_Lipsync.ipynb

**三．各个技术框架介绍**

FFmpeg是领先的多媒体框架，能够解码、编码、转码、多路复用、解复用、流式传输、过滤和播放人类和机器创造的几乎所有内容。它支持最晦涩的古代格式，直至达到最前沿。无论它们是由某个标准委员会、社区还是公司设计的。它还具有高度的可移植性：FFmpeg在各种构建环境、机器体系结构和配置下，在Linux、Mac OS X、Microsoft Windows、BSD、Solaris等平台上编译、运行并通过我们的测试。

https://zhuanlan.zhihu.com/p/592988688

https://github.com/FFmpeg/FFmpeg

Whisper是一个开源的自动语音识别系统，它在网络上收集了680,000小时的多语种和多任务监督数据进行训练，使得它可以将多种语言的音频转文字。

Whisper的好处是开源免费、支持多语种（包括中文），有不同模型可供选择，最终的效果比市面上很多音频转文字的效果都要好。

https://zhuanlan.zhihu.com/p/618598430

https://github.com/openai/whisper

Googletrans是一个免费和无限制的python库，实现了谷歌翻译API。使用Google Translate Ajax API来调用诸如检测和翻译之类的方法。

https://github.com/ssut/py-googletrans

TTS全称Text-to-Speech，翻译过来叫做文本生成音频。这种方式大家比较熟悉，抖音上，有许多应用TTS技术的视频。比如配着大叔口音的搞笑视频，再比如主人公叫小帅、小美的三分电影解说。特点就是声音带着机械感，可选模板比较少。

https://baijiahao.baidu.com/s?id=1765686678971119247&wfr=spider&for=pc

https://github.com/coqui-ai/TTS

So-vits-svc,，SVC（Singing Voice Conversion），歌声转换，也就是类似变声器的玩意，抽取一个人的声音作为训练数据，训练一个神经网络模型，学习他的声线；然后用模型在目标歌曲上做推理，即可实现用自己的声线唱目标歌曲。

https://zhuanlan.zhihu.com/p/626164250?utm\_id=0

https://www.bilibili.com/read/cv23788651/

https://github.com/svc-develop-team/so-vits-svc

VideoReTalking是一个新系统，可以根据输入音频编辑现实世界中会说话的头部视频的人脸，即使在不同的情绪下也能产生高质量的假唱输出视频。系统将这一目标分解为三个顺序任务：

（1） 基于正则表达式的人脸视频生成

（2） 音频驱动的唇同步和

（3） 面部增强，以提高照片的真实感。

<https://github.com/OpenTalker/video-retalking>

Audio Separator是一个Python包，允许使用训练的ONNX格式模型将音频文件分离为两个词干，一个是vocal词干，另一个是instrument词干。

<https://github.com/karaokenerds/python-audio-separator>

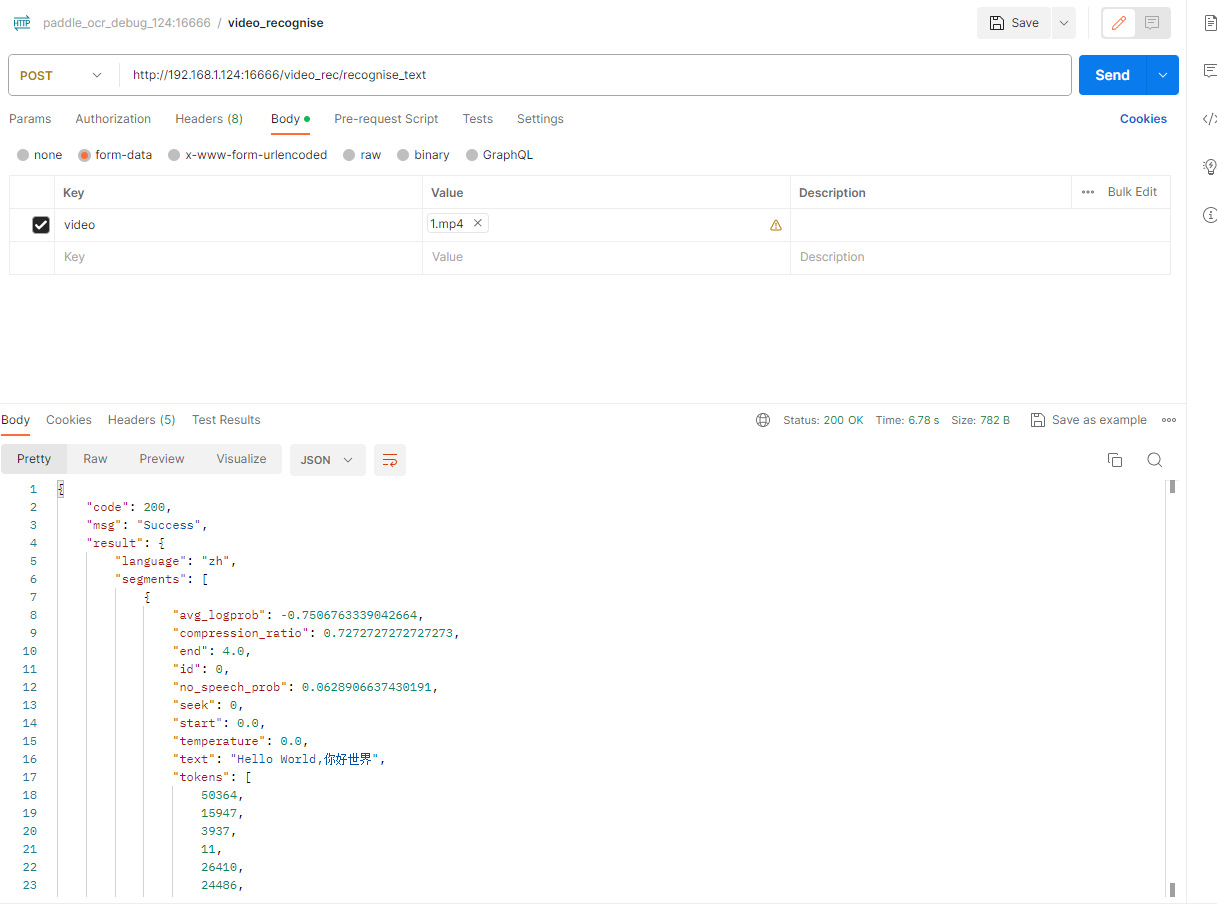
PaddleSpeech是PaddlePadle百度飞桨深度学习框架下的语音服务工具包。PaddleSpeech包括自监督学习模型、带标点符号的 SOTA/流式 ASR、带文本前端的流式 TTS、说话人验证系统、端到端语音翻译和关键字识别。

http://wed.xjx100.cn/news/200818.html?action=onClick

<https://github.com/PaddlePaddle/PaddleSpeech>

**四．当前成果**

经过研究，已经成功实现了视频提取音频并转换为文本，可以通过api借口进行调用，通过入参传入mp4或wav等主流视频格式即可将视频内的文本提取出来以json格式返回。



通过tts与paddle speech进行文本语音合成，由于依赖环境差距过大无法直接置于当前环境下使用需要额外配置隔离运行，仅在测试环境下进行了尝试，成功生成了音频。

通过so-vits-svc实现了对录制音频与tts生成音频的音源替换，利用audio separator将歌曲的人声部与伴奏部进行分离，并使用so-vits-svc实现让特朗普来唱中文歌。

**五．面临的困难与下一步计划**

1. 由于各个技术框架间的依赖环境差异过大，无法在同一个环境配置下使用，导致需要开启多个服务才能实现对应功能，从而影响了功能间的整合与交互。

设想通过开启多个服务进行服务间的调用，从而实现整套链路的闭环，相对应的要处理好多线程并发可能带来的问题。

1. tts与paddle speech可用模型数量较少，paddle speech功能偏向与单独生成中文或英文的音频，语言混合效果较差。tts主要侧重于英语音频生成，对中文的优化效果有待开发。可以再寻找其他模型或技术来代替，或者将英文先全部翻译为中文再进行单一语言的音频生成，声音效果可以通过svc来更改音源。
2. 要对文本生成的音频与视频进行一对一的适配，在正确的时间点插入音频，这项难点有待研究。预计可行的方案为将对话按照间隔拆开，对单一音频进行操作，最后根据顺序拼接，但若生成音频时间长于原音频则会产生错位，可以考虑加速以缩短时间。

**六．可能实现的功能与服务**

1. 视频转换为音频
2. 视频、音频转换为文本
3. 视频、音频转换文本并翻译
4. 视频、音频转换文本翻译后进行tts音频合成，或根据svc进行音源转换
5. 根据文本通过tts进行音频合成，或根据svc进行音源转换
6. 音频通过svc进行音源转换
7. 歌曲的伴奏与人声提取
8. 歌曲的svc换声
9. 视频音频文本翻译后tts生成音频，进行svc音源转换后再通过对嘴形合成为新视频
10. 通过文本总结ai将视频提取的文本进行内容提取与总结
11. 将伴奏通过ai转换为曲谱，便于演奏
12. etc...