武汉大学计算机学院 2022级中级项目实训周报

**专业班级：** 计卓班 **学号：2022302111279** **姓名：邱涵**

**项目名称： 缤果数智 小组成员：邱涵、陈震雄、陈晋宇、李佳隆**

**时间阶段：**第 1 周， 2024 年 7 月 1 日至 7 月 6 日

1. **本周计划和项目任务**

1、完成基本分工(明确为前端、后端和算法组)、明确前后端搭建技术栈和核心算法技术

2、(算法组)完成Wav2Lip288x288的基本搭建，读懂Wav2Lip288x288的代码逻辑，用该算法训练基本模型

3、(算法组)构建Wav2Lip288x288的训练数据集，下周开始用自有训练集训练数据

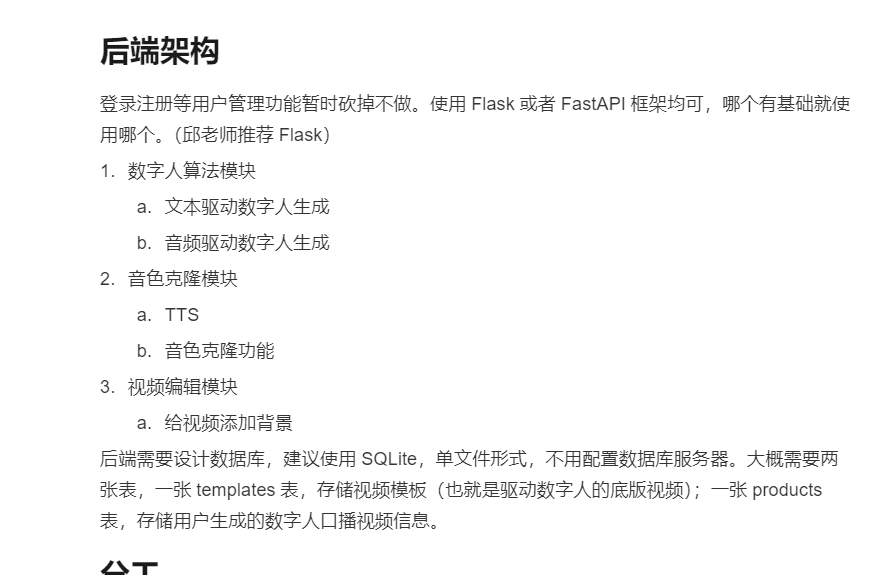
4、(后端组)学习后端Flask技术栈的使用和运用，下周开始实际搭建

5、(后端组)明确后端搭建所需模块，完成后端的实际分工

1. **本周进度/任务完成情况**

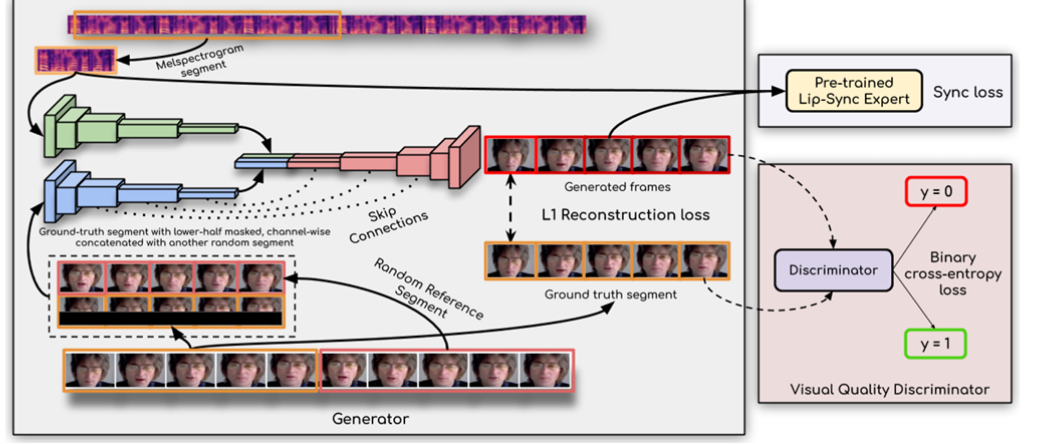
1、 完成了基本分工，成功分好了前端组和后端组和算法组。

2、明确了前端技术栈主要为Vue3，后端技术栈主要为Flask、SQLite。我需要负责的核心算法模块主要是Wav2Lip 288x288

3、后端的主要模块如下:

4、学习了Wav2Lip288x288的基本流程、读懂了代码和论文：

论文地址: <https://arxiv.org/abs/2008.10010>



主要就是分成**音频处理层、图像处理层、唇形生成层、后处理层**，各层功能如下：

音频处理层：主要是**音频特征提取**：这一层负责从输入的音频信号中提取有用的特征，如梅尔频率倒谱系数（MFCC）。这些特征用于表示音频的时间和频率特性，是后续唇形预测的基础。

音频预处理：包括归一化、降噪等步骤，以确保音频数据的质量和一致性。

图像处理层：首先是**面部检测与对齐**：这一层使用面部检测算法（如Dlib或OpenCV）来定位视频帧中的面部区域，并进行对齐，以便后续处理。然后是**关键点检测**：提取面部的关键点（如眼睛、鼻子、嘴巴等），这些关键点用于指导唇形的生成。

唇形生成层：主要是**生成网络**：这是一个深度学习网络，通常基于卷积神经网络（CNN）或变分自编码器（VAE），它接收音频特征和面部图像作为输入，输出预测的唇形图像。，然后这里的**损失函数**定义了生成网络的训练目标，如像素级别的重构损失、对抗损失等，以确保生成的唇形既准确又自然。

后处理层：主要是**融合与平滑**：将生成的唇形与原始视频帧进行融合，使用图像处理技术（如泊松融合）来确保过渡自然，减少视觉上的不连续性。同时包括**质量增强**：可能包括颜色校正、光照一致性调整等，以提高最终视频的质量。

代码主要运行两个训练代码，首先是train\_syncnet\_sam.py，训练syncnet层，这一部分就是确保能根据输入的音频生成对应的唇形，主要侧重于两者的匹配性。然后是hq\_wav2lip\_sam\_train.py，这个就是生成实际的面部驱动，侧重生成的唇形能够和面部融合。

5、我构建了训练所需的数据集，并构建了预处理脚本。由于由于较少数据集就能取得较好的效果，所以我号召同学好友，构建了大约20人的数据样本，同时每个人的数据样本又由5-6段10s左右的短视频构成。预处理过程中，原版wav2lip的输入视频会被切割成96x96大小的短图片，为了保留更多的特征，我决定将输入的视频切割成288x288的大小，也即Wav2Lip288x288，通过随机采样得到足够多的288x288大小的图片(.jpg)，同时切割音轨输出得到对应的音频(.wav)后(通过名字和编号构建音频-图片的一一对应关系)，就得到了输入网络的最终训练数据集。

6、后端部分，下周进行具体分工，这周主要是学习了Flask的基本用法，参考教程:

【快速上手 — Flask中文文档(3.0.x)】https://dormousehole.readthedocs.io/en/latest/quickstart.html

1. **下周的计划和项目任务**

1、后端三个模块具体分工，准备好开发规范。

2、完成我负责的后端模块，大概率是数字人算法模块。文本驱动和音频驱动部分，需要构建对应API接口。

3、模型继续训练，目前效果不理想，同时显存有限，收敛速度较慢，可能需要向老师申请算力。

1. **存在的主要问题或特殊情况**

主要问题是有一些组员由于竞赛、身体状况等原因，没有办法及时到场，很多交流都是线上交流，比较吃力，分工安排，进度交流效率较低，这一周需要注意。