**单人组队自评（0-1.0）**

**1.3.1自评系数**

**0.7**

**1.3.2遇到困难及自评系数的依据**

**困难：**

1.虚拟环境配置

首先是虚拟环境的配置，在windows运行时，我们首先要使用conda创建虚拟环境，这里需要选择python=3.8版本，接着我们需要安装wav2lip项目中需要的库以及包，当我直接按照项目中提供的requirement.txt进行下载时，出现了报错，所以这里我们对一些包需要选择正确的版本，例如librosa需要选择0.8.0版本，项目中提供的numpy版本过低，但是版本过高的话也会导致后续代码无法运行，所以这里需要选择1.23.3版本，这里要注意的是不管是在windows还是在docker中都需要cuda=11.8版本，不然在评估时会出现报错。同时，在评估方法中我们用到了依赖于syncnet网络的LSE-C和LSE-D分数评估方法和pytorch\_fid评估方法，我们需要额外下载需要的库。

2.实验结果生成

在结果生成中遇到的问题分为生成视频结果和评估结果两方面。首先是生成视频中遇到的问题，在该模型中，我们可以选择图片或者视频作为模型的人脸输入，当我们使用视屏作为人脸输入时，如果用cpu运行模型，运行程序时间会比较长，并且对于wav2lip\_gan模型，可能会出现程序卡死的情况，此时有两种方法，一种是通过修改inference.py文件中的bitch\_size，我们需要限定块的大小（最小值为1），另一种方法就是我使用的，在windows下我们可以通过conda install pytorch torchvision torchaudio pytorch-cuda=11.8 -c pytorch -c nvidia命令来安装cuda11.8，此时windows下就可以自动调用gpu来运行模型，而在docker中，我们需要提前在docker run阶段设置好gpu运行。另一方面是关于评估结果生成的问题，我在使用syncnet的无监督评测方法时发现需要自行下载模型，这个在原项目中我没有找到，所以从网上下载了预训练的模型。使用pytorch\_fid方法，本质上是将两个文件夹的图像进行比对生成FID指数，所以这里需要我们先自行 编写代码将生成视频和输入视频按照帧提取为图像，再通过python -m pytorch\_fid比较两个文件夹，这一步最大的困难是windows和docker下运行起来不相同，在docker运行该命令时，会报错显示共享内存不够用，我首先尝试用--shm-size=3g扩大docker的共享内存为3g，但依旧报错，在上网查阅资料后，在命令后添加--num-workers 0 --device cuda:0，程序正常运行。

3.docker封装

这一部分遇到的问题是我在本次实验中遇到问题最多的一部分，因为在此之前我并没有使用过docker，所以首先遇到的问题就是如何使用docker，在docker中拉取镜像时，因为网络问题会显示无法拉取，所以此时需要我们通过国内的镜像网站来进行拉取，同时还有在工作区域内挂载我们要需要用到的目录，为了确保模型正常运行，需要在docker run阶段输入--gpus all和--shm-size=3g设置共享内存和gpu运行。

**自评系数的依据**

对于0.7这个自评系数，我觉得0.6是基于项目难度的基础分，而剩下的0.1是基于我能够单人组队并最后成功完成这项实验的奖励分。