

# ReadMe: Text - to - LipSync

## To work with the Wav2Lip model :

1. Clone this repository.
2. Install the following requirements.
  - a. tensorflow tensorflow-gpu
  - b. librosa==0.7.0
  - c. numpy==1.17.1
  - d. opencv-contrib-python>=4.2.0.34
  - e. opencv-python==4.1.0.25
  - f. torch==1.1.0
  - g. torchvision==0.3.0
  - h. tqdm==4.45.0
  - i. numba==0.48(As also mentioned in the requirements.txt)
3. Ensure you have the “.pth files” (Pretrained models) in the respective folders.
  - a. /Wav2Lip/checkpoints/
  - b. Wav2Lip/face\_detection/detection/sfd/If you don't have these in the mentioned folders try downloading them from the following links and place them in the correct folder.  
[https://drive.google.com/drive/folders/1cxf6ywJZGWX\\_fuFbgxTCEDiLn-BJgr9](https://drive.google.com/drive/folders/1cxf6ywJZGWX_fuFbgxTCEDiLn-BJgr9)  
<https://www.adrianbulat.com/downloads/python-fan/s3fd-619a316812.pth>
4. To run this model we need to run the inference.py file with some arguments as we see in the following command

```
cd Wav2Lip && python3 inference.py --checkpoint_path
checkpoints/wav2lip_gan.pth --face
"/Users/jatin2412/Desktop/Wav2Lip/static/uploads/input_video.mp4" --audio
"/Users/jatin2412/Desktop/Wav2Lip/static/uploads/input_audio.wav"
```

Make sure that input videos are available in the static/uploads folder and then change the above-mentioned path as per the system on which the model is being run.

5. You will get the output “.mp4” file in the static/results folder.

You may also refer to the following links to test it on google colab or check the original project where one can also explore the training part of the same:

- a. <https://github.com/Rudrabha/Wav2Lip>
- b. <https://colab.research.google.com/drive/1ljFW1cLevs6Ouyu4Yht4mnR4yeuMqO7Y#scrollTo=MH1m608OymLH>

We used FLASK to create APIs and an interface, where all the user would have to do is upload the input audios and videos and play the final .mp4 file.

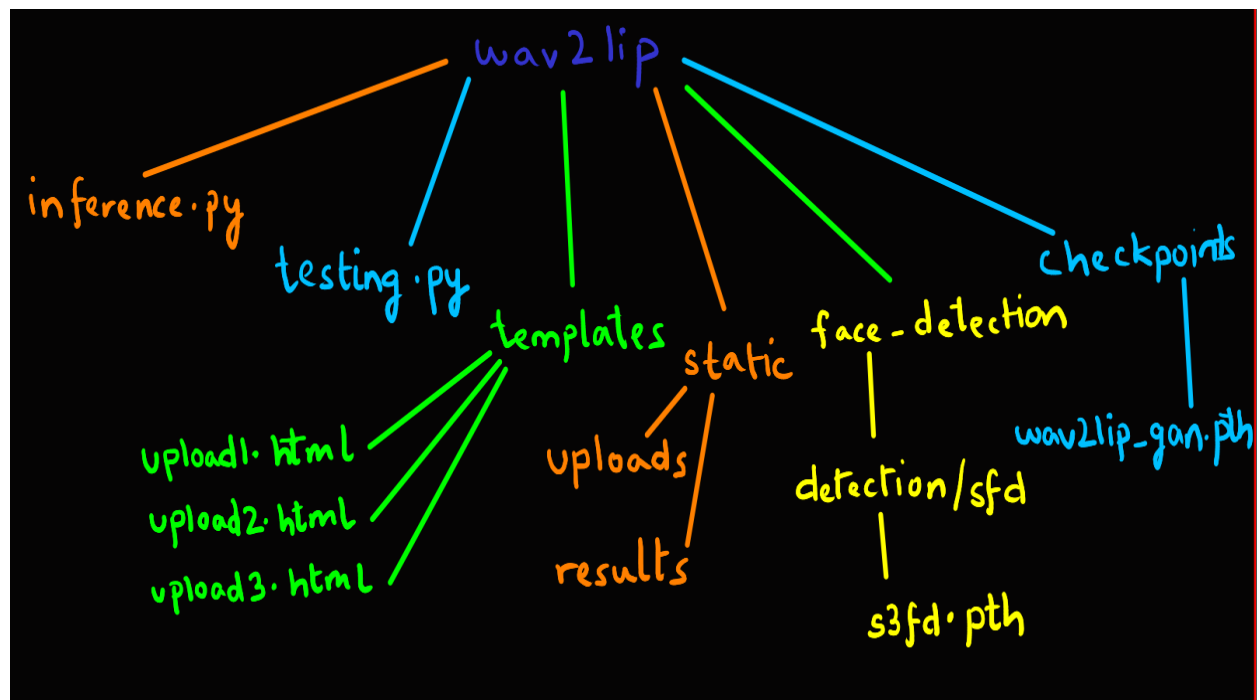
### **To run this flask application:**

1. First, make sure you have the above-mentioned Wav2Lip model working in your system.
2. Install flask-restful. You may use pip, brew etc in order to do so.
  - a. For eg. pip install Flask, sudo pip3 install flask-restful
3. When you run the application the video and audio files uploaded are saved in the static/uploads folder.
4. In the testing.py file(line 48) make sure you change the audio and video input paths as per the system in which the application is being tested.

```
call(["python3", "inference.py" ,  
"--checkpoint_path","checkpoints/wav2lip_gan.pth","--face" ,  
"\<path>/Wav2Lip/static/uploads/input_video.mp4\"","--audio", "\<path>/Wav2L  
ip/static/uploads/input_audio.wav\""])
```

5. The inference.py file after running will give the output in the static/results folder which will then be played by the flask application.

## File Structure:



## Working in a nutshell:

