EE 599 Project Report – Speech Animation

By: Brinal Bheda, Chaitanya Gupte, Manasa Manohara, Shruti Gadewar

Introduction:

To experiment with a deep learning approach to automatically animate mouth texture using the speech input. The range of practical applications includes generating high quality videos from audio by significantly reducing the amount of bandwidth needed in video coding. This could be extended to enable lip-reading from over-the-phone audio which could help the hearing-impaired community. Previous work has required the input to be heavily constrained with the subject specifically lit and made to say phonetically rich sentences. Our method can be generalized to online interviews, video calls without being extremely constrained.

Problem statement :

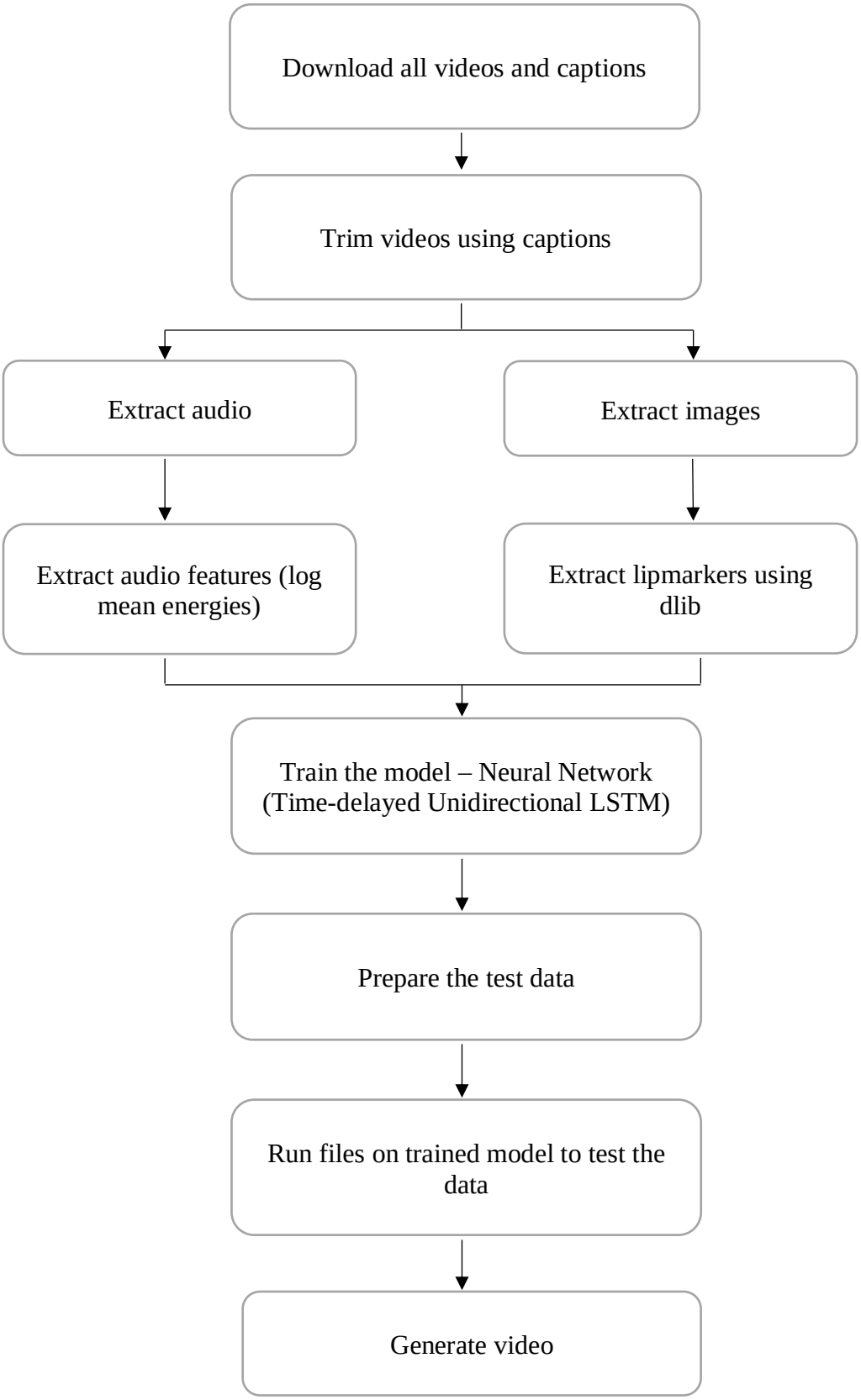
1. To experiment with deep learning approaches to automatically generate mouth texture animations to synchronize with the input speech.
2. The range of practical applications includes generating high quality videos from audio by significantly reducing the amount of bandwidth needed in video coding.
3. This could be extended to enable lip-reading from over-the-phone audio which could help the hearing-impaired community.

Procedure:

1. Trim the videos using captions.
2. Take the trimmed video and separate audio and image components.
3. We use input audio, extract the audio features. Input is converted into a sequence of Log Filter bank energies.
4. Extract the facial features for lip marking using dlib.
5. Upsample the lip markers and apply PCA.
6. We represent the audio using standard Log Mel Filter bank energies and mouth shape by 40 lip markers reduced by a PCA basis.
7. Use an LSTM neural network on audio features.

8. Synthesize the video from the test data.

Flowchart:



Implementation:

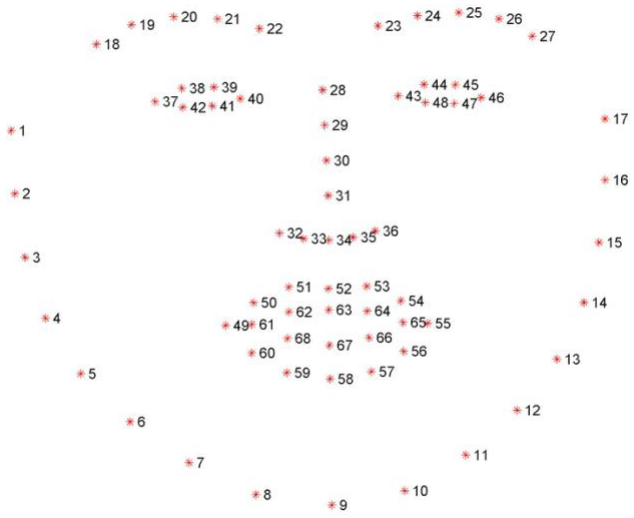
The steps for the implementation are as follows:

- Data Preprocessing

The Dataset used for our implementation mentioned in "Synthesizing Obama - Learning Lip Sync from audio" paper and the link for the same is given below:
http://grail.cs.washington.edu/projects/AudioToObama/obama_addresses.txt
It contains 303 weekly addresses: Each video is about 3 minutes which sums up to total 17 hours of video. Videos were trimmed caption to caption using WebVTT. The audios were extracted from the trimmed videos at 16kHz and mono-channel. The videos were cropped to have Obama in focus and then frames were extracted at 30 fps. The frame size for the images extracted was 256x256.

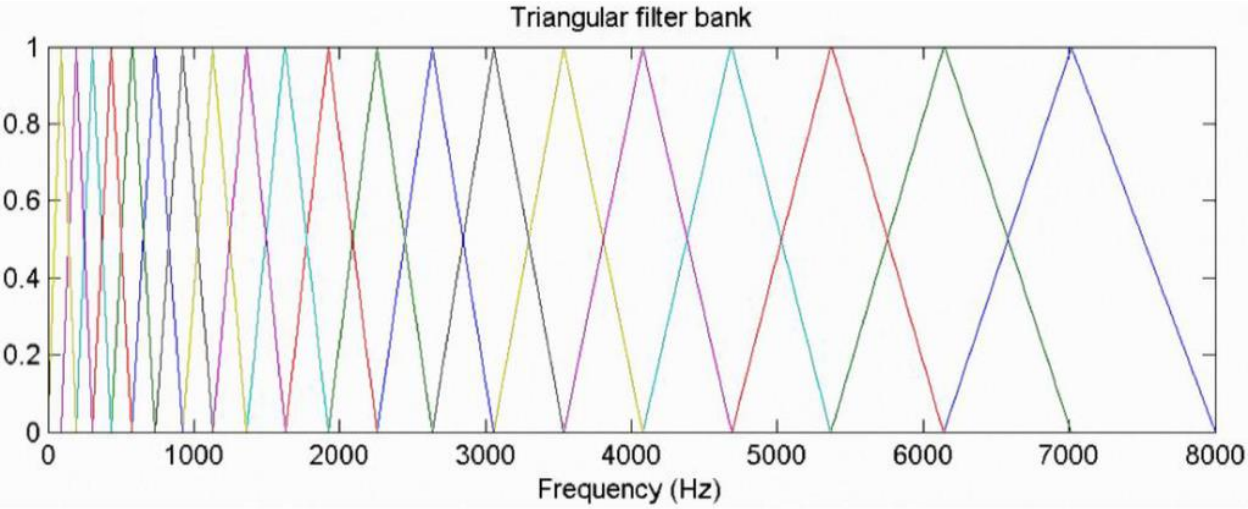
- Video features Extraction

Using OpenCV's dlib library, facial landmarks were extracted using the frontal face detector. The obtained keypoints were of shape (68,2). The extracted keypoints were then mean centered. Tilt was calculated to place the mouth landmarks at an angle and in accordance with the face. Tilt of the head is obtained using the eye landmarks extracted. The keypoints are normalized to be scale-invariant.
Normalized lip markers, normalization factor, tilt, mean, normalized keypoints, and original keypoints are dumped in a pickle.



- Audio features Extraction

The feature considered for audio feature extraction was log Mel filter bank energies.
Parameters used for audio extraction were as follows:
Number of filter banks = 26.
Window length = 25ms
Window step = 10 ms
FFT size = 512
The length of the audio and the upsampled video size is asserted to be equal.



- Video features extraction

OpenCV is used for frontal face detector to obtain the facial landmarks – shape (68,2). The extracted keypoints are mean centered. Tilt of the head is obtained using the eye landmarks extracted. The keypoints are normalized to be scale-invariant. Normalized lip markers, normalization factor, tilt, mean, normalized keypoints, and original keypoints are dumped in a pickle.

- Mouth Shape features

To compute the mouth shape representation, we first detect and frontalize Obama's face in each video frame using the approach. For each frontalized face, we detect mouth landmarks using which gives 40 points along the outer and inner contours of the lip. We reshape each 20-point mouth shape on the X and Y axis, apply PCA over all frames, and represent each mouth shape by the coefficients of the first 20 PCA coefficients; this step both reduces dimensionality and decorrelates the resulting feature set. Finally, we temporally upsample the mouth shape from 30Hz to 100Hz

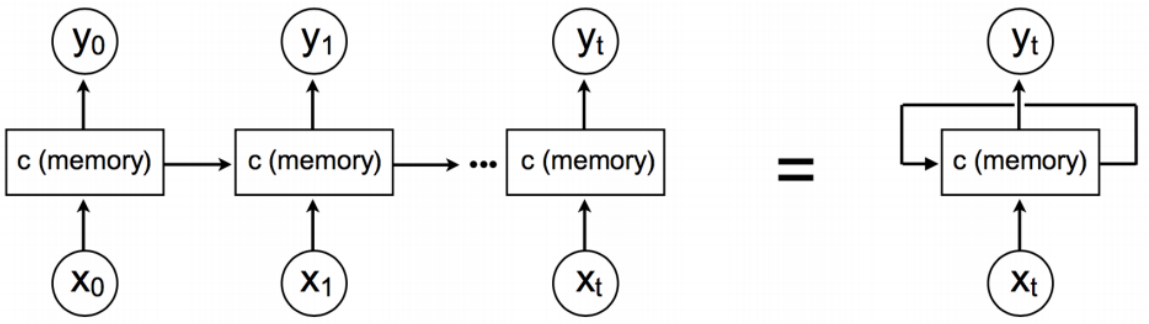
by linearly interpolating PCA coefficients, to match the audio sampling rate. This upsampling is only used for training; we generate the final video at 30Hz.

- PCA and Upsampling

40-dimensional mouth features were reduced to top 20 features. PCA parameters were pickled.
To assure that the keypoints had a continuous flow and to match the fs of audio samples, the extracted frames were upsampled. The parameters used to upsample the mouth features : Look_back = 50 samples; time_delay = 20 samples.
The keypoints were estimated and overwritten considering the previous keypoints and the future keypoints.

- Long Short Term Memory (LSTM)

The network consists of a Unidirectional LSTM with time delay. The network consists of 60 LSTM nodes. Uses a 50-step time-delay (corresponding to 200ms). Trained for 30 epochs with a batch size of 60. Best weights are saved and reloaded for a set of 50 videos at once because of RAM limitations. We use the ADAM optimizer. Implemented in Keras with Tensorflow backend. The output size is 20 that is dimension of the reduced mouth features using PCA. This helps to learn a mapping from MFCC audio coefficients to PCA mouth shape coefficients. This network takes the latest audio input x_t, uses it to modify its hidden state, memory c, and outputs a new mouth shape vector y_t for that time instant, as well as passing its memory forward in time.



- Training and Testing

Using audio KP, video KP, PCA pickle files.

1. Create audio samples such that both future and past features are considered to predict the lip markers of a video frame.
2. This was done to implement time delay in unidirectional LSTM.
3. 80% for training, 10% for validation, 10% for testing.
4. X = array of extracted audio features
5. Y = mouth feature keypoints of a single frame.

6. Obtained Y prediction is rescaled and inverse PCA is applied to get back all the 40 components.
7. For testing on an unseen video - the same video pre-processing technique was applied and the predicted keypoints were overlapped on the extracted frames.
8. **Loss obtained = 0.0182**

Complexity Analysis:

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm_1 (LSTM)                (None, 60)                20880
_________________________________________________________________
dropout_1 (Dropout)          (None, 60)                0
_________________________________________________________________
dense_1 (Dense)              (None, 20)                1220
=================================================================
Total params: 22,100
Trainable params: 22,100
Non-trainable params: 0
```

Existing Codebase and Papers:

https://github.com/supasorn/synthesizing_obama_network_training
https://github.com/mrmotallebi/synthesizing_obama_network_training

Major Challenges:

1. Dealing with a huge dataset -- downloading videos and extracting features was time consuming.
2. Understanding how 'Tilt' of the head could affect the movement of the lips.
3. Understanding how PCA was applied to the mouth features.
4. Logic to upsample the frames to maintain the flow of the video.
5. Audio and video were not of same lengths.
6. Training took a long time. (Still training :P)

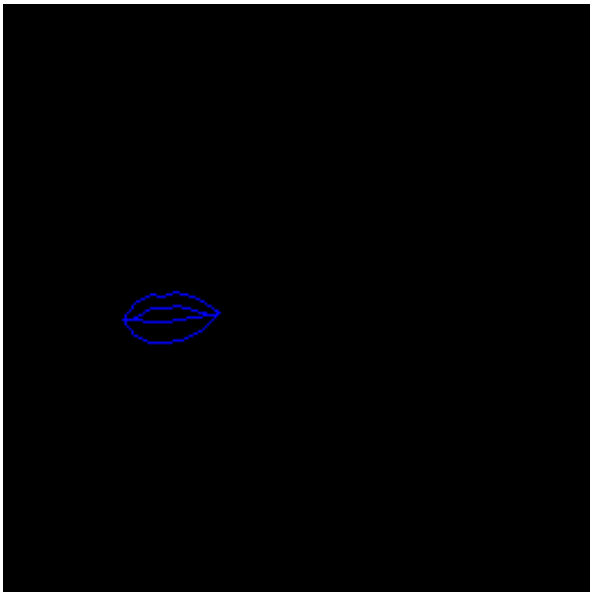
Adaptation from Original Plans:

1. In the original paper, 18 lip markers were extracted from the images, but we used dlib which extracts 20 lip markers.

Individual Contributions:

1. BRINAL: Audio feature extraction, test data preparation, training, requirements and config.
2. CHAITANYA: Video trimming and feature extraction, PCA and upsampling, model experiment.
3. MANASA: Video trimming and feature extraction, training the model, refactoring.
4. SHRUTI: Audio feature extraction, PCA and upsampling, training, model experiment, test data preparation.
5. Common tasks: Downloaded data and co-ordinated pickling of files.

Results:



Application:

- Make the network speaker independent.
- Character animation – Combine the rig shapes and mouth texture to create a new character.
- Speech Summarization – Given a long address speech, create a short summary version by manually selecting the desired sections from transcribed text. Our method can be used to generate a seamless video for the summarized speech.
- Generating high quality videos from audios by reducing bandwidth required for video transmission.
- Enable lip reading over the phone audio to help the hearing-impaired community.

Citations and References:

- Supasorn Suwajanakorn, Steven M. Seitz, and Ira Kemelmacher-shlizerman, Synthesizing Obama: Learning Lip Sync from Audio https://grail.cs.washington.edu/projects/AudioToObama/siggraph17_obama.pdf
- Sarah Taylor, Taehwan Kim, Yisong Yue, Moshe Mahler, James Krahe, Anastasio Garcia Rodriguez, Jessica Hodgins, Iain Matthews, A Deep Learning Approach for Generalized Speech Animation
- Yang Zhou, Zhan Xu, Chris Landreth, Evangelos Kalogerakis, Subhransu Maji, Karan Singh, VisemeNet: Audio-Driven Animator-Centric Speech Animation
- Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. Neural computation, 9(8): 1735–1780, 1997.
- Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. arXiv preprint arXiv:1611.07004, 2016.
- Tero Karras, Timo Aila, Samuli Laine, Antti Herva, and Jaakko Lehtinen. Audio-driven facial animation by joint end-to-end learning of pose and emotion. ACM Trans. Graph., 36(4):94:1– 94:12, July 2017. ISSN 0730-0301. doi: 10.1145/3072959.3073658. URL http://doi.acm. org/10.1145/3072959.3073658.
- Jose Sotelo, Soroush Mehri, Kundan Kumar, Joao Felipe Santos, Kyle Kastner, Aaron Courville, and Yoshua Bengio. Char2wav: End-to-end speech synthesis. 2017.
- Supasorn Suwajanakorn, Steven M Seitz, and Ira Kemelmacher-Shlizerman. Synthesizing obama: learning lip sync from audio. ACM Transactions on Graphics (TOG), 36(4):95, 2017.
- J. Thies, M. Zollhöfer, M. Stamminger, C. Theobalt, and M. Nießner. Face2face: Real-time face capture and reenactment of rgb videos. In Proc. Computer Vision and Pattern Recognition (CVPR), IEEE, 2016.