Single Audio to Inferred Animation

PROGRESS REPORT 1

COMP 400

Kevin Junyang Cui (260984715)

supervised by

Joseph Vybihal

McGill University
School of Computer Science

May 31, 2023

PROJECT DESCRIPTION

The virtual avatar, Virtual YouTuber, or *VTuber*, is a phenomenom in recent cultural spheres in which an entertainer's motion or likeness is represented entirely by an animated character, either in 3D or 2D (e.g. *Live2D*[1]), for the purpose of both real-time streaming and animation playback. While the primary method for high-fidelity real-time motion-tracking has been via either video (such as with computer vision), infrared camera, or wearable devices, these methods prove to be a challenge both in terms of affordability and accessibility. Meanwhile, alternatives such as keyboard-based interfaces impose limitations on precision and practicality. Audio or speech-based tracking has become available in some tools such as *VTube Studio*, which employ rudimentary acoustic analysis for the purpose of optimising lip-synchronisation in addition to video facial tracking, based on attributes of the inputted microphone audio such as volume and phonemes detected in the frequency domain.[3] Novel approaches such as *AlterEcho* have attempted to further infer movement such as non-verbal behaviour from audio input, including text-to-speech and acoustic analysis (which map to a repertoire of gestures).[7] This allows not only for a more affordable, accessible interface with no additional cognitive effort on the subject's part, but also opens the door to a more efficiently encoded input and added personalisation of the output via manually inputted static parameters through loosened coupling between the subject and the avatar. However, such novel methods are limited by factors such as language spoken, latency, and granularity of personalisation. This project will attempt to use a neural network to more robustly infer non-verbal behaviour for animation from a single audio input. Similar to models that train for audio to facial animation blendshapes[8] or emotional classifications[6], this project will use features in the frequency domain extracted from audio such as Mel-frequency cepstral coefficients (MFCC) to yield parameters that encode upper-body animation, such as 3D or Live2D parameters. The project aims to empirically train on different feature extraction methods and architectures to find a model that optimises for latency, naturalness of movement, and granularity of personalisation.

STORYBOARD

0.1 ARCHITECTURE

0.1.1 Audio Feature Extraction

It can be hypothesised that the most relevant features in audio for animation (assuming speech in an arbitrary language) will be phonetic and tonal. That is, all features would be derivable from the MFCC of the target audio.[6][8] It can be noted that LPC is another possible avenue of feature extraction, but is observably inferior.[8] It may be advantageous to also further extract phonemes (or visemes) from the MFCC, as more precise additional features, with the hypothesis that there exists some strong relation between motion of the lips and upper body.

0.1.2 Neural Network Training

A bidirectional LSTM with an embedded attention mechanism has been shown to be effective in training for audio to facial animation, as it gives the network memory of past input audio features.[8] Therefore, it would follow that a similar model may be effective in inferring upper-body animations. As it can be assumed that upper-body animations may require more nondeterministic inference dependent on the subject, the model may need to be adjusted to also take in some static abstract parameter representing the personality of the subject (persona parameters[7]) which may need to be determined empirically. Training may require self-generated data, with isolated audio mapped to a n-dimensional record of n motion parameters captured via conventional motion tracking. Each record of parameters can be used to compute the loss from its corresponding windowed excerpt of the MFCC (such as with discrete-time Short-term Fourier Transform). There may be difficulty in generating a dataset of sufficient size.

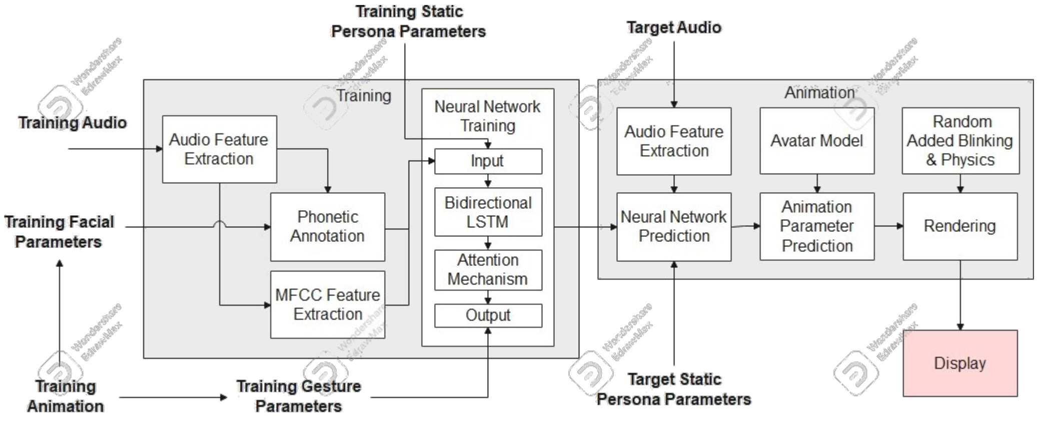

Figure 1: Sample architecture of project

0.1.3 Animation

Once trained, the output of the model given some target audio input should be an n-dimensional record of n motion parameters, either in 3D or Live2D. Parameters are directly applied to a model in an engine. Additional random blink patterns (on top of inferred blinking) and physics may be optionally applied. The rendered video is streamed or saved.

0.2 DELIVERABLES

End of Semester Statement 1 *By the end of the semester, this project aims to be able to clearly and consistently infer some upper-body animation from single audio, in some way which is indistinguishable from conventional motion-capture to the common viewer.*

End of Semester Statement 2 *By the end of the semester, given enough time, this project aims to be able to clearly and consistently infer complete upper-body animation from single audio, possible in real-time, with adjustable parameters for personalisation. The goal is to output motion that complements the audio, as opposed to strict coupling as in conventional motion-capture[7], which may require novel approaches to audio feature extraction and recurrent neural network architecture.*

Table 1: Deliverable dates

Date	Description of Task	Credits
15 June	Configure runnable base MFCC extraction and LSTM	0.5
22 June	Modify network to apply gesture parameters to loss function	0.5
29 June	Develop interface and recorder for training dataset	0.5
6 July	Collect data and create dataset of audio and parameters	0.5
13 July	Develop animation engine and base lip-sync	0.5
3 August	Train usable model for audio to animation inference	0.5
17 August	Train model which passes established test case(s)	0.5
24 August	Final project and report	0.5

0.3 TEST CASES

0.3.1 Model performance test case

Quality of the model can be evaluated against the ground truth values. Held-out animation video (of at least 2 minutes length with at least 2 actors) should be acceptably plausible. Quantitative evaluation can be carried out via testing the performance against different numbers of nodes and types of models. One type of quantitative evaluation is using the root mean squared error (RMSE) of parameters over all frames in a held-out test set[8]. Success of an architecture can thus be evaluated as achieving a lower RMSE than all other architectures.

$$\varepsilon = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - y_i^p)^2}$$

0.3.2 Animation performance use case

Naturalness and engagement can be measured via user study, which compares this project to other VTuber tools such as conventional motion capture, keyboard-based animation such as *VMagicMirror*, and possibly AlterEcho.[7] This would follow closely to the evaluation performed by the AlterEcho team. The objective would be to replicate the survey done by the AlterEcho team ($N \geq 315$) with VTubers and other streamers to acheive higher ratings on a five-point Likert scale on naturalness (4.02 ± 1.04) and engagement (4.16 ± 0.89).

0.3.3 Run-time performance test case

An additional test case for success would be the exact latency of running the project alongside some streaming software, in comparison to other VTuber tools.

0.4 TECHNOLOGY STACK

The project will use PyTorch, possibly with Adam[4] for optimisation. Phonetic annotation (if used) can be completed using CMUSphinx[5] or Wav2Vec2[2]. The entire framework may be implemented in Python or Jupyter Notebook. Animation may be rendered via Blender (3D) or Unity (3D or Live2D).

REFERENCES

[1] Live2d cubism. https://www.live2d.com/en/. Accessed 2023-05-30.

[2] Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations, December 2020.

[3] DenchiSoft. Vtube studio settings. https://github.com/DenchiSoft/VTubeStudio/wiki/VTube-Studio-Settings, July 2022. Accessed 2023-05-30.

[4] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization, December 2014.

[5] Paul Lamere, P. Kwok, E. Gouvêa, Rita Singh B. Raj, William Walker, Manfred K. Warmuth, and Peter Wolf. The cmu sphinx4 speech recognition system, January 2003.

[6] R Raja Subramanian, Yalla Sireesha, Yalla Satya Praveen Kumar Reddy, Tavva Bindamrutha, Mekala Harika, and R. Raja Sudharsan. Audio emotion recognition by deep neural networks and machine learning algorithms, October 2021.

[7] Man To Tang, Victor Long Zhu, and Voicu Popescu. Alterecho: Loose avatar-streamer coupling for expressive vtubing, October 2021.

[8] Guanzhong Tian, Yi Yuan, and Yong Liu. Audio2face: Generating speech/face animation from single audio with attention-based bidirectional lstm networks, July 2019.