RO031

Sign2Speak! Synthesising Emotional Speech from Singapore Sign Language through Deep Learning

Report

1 INTRODUCTION

Sign language is a form of communication used by deaf individuals worldwide. Just like spoken languages, there are over 200 region-specific signed languages, from American Sign Language (ASL) to Singapore Sign Language (SgSL) and many more. With more than 70 million deaf individuals worldwide relying on sign language as their first language, the communication barrier between signers and the larger non-signer community is a growing concern due to the high dependency on translators to facilitate conversation. In Singapore, as of 2021, there are 6000 SgSL users with only 9 official and 20 community interpreters, posing a challenge to the independence of the deaf community in their day-to-day lives. They are also costly and require a lot of paperwork. [1],[2],[3]

Since the pandemic, many have resorted to calls to carry out their activities (i.e., medical checkups). Those relying on Sign Language may struggle to do so, which leaves doctors unaware of their patients' feelings, an important factor in diagnosis. Having an interpreter invades their privacy, and translation or written means of communication is a slow process. Further, interpreters may not accurately capture the emotion signers wish to express.

As such, this research project aims to take a more inclusive approach to bridge communication between both parties, aligning itself with Singapore's objective of inclusion of those with special needs, as discussed in the recent Forward SG exercise launched in June 2022. [4],[5]

As its first step, this study focuses on a better alternative to speech synthesis from sign language that can rival existing proposals in terms of non-intrusiveness and accuracy. In the long run, this could reduce one's overreliance on translators and increase the deaf community's independence, privacy and integration into modern society. The findings can be used in making applications for smoother communication in day-to-day scenarios and teleconferencing.

Emotion is paramount in daily communications. In fact, the manner in which something is said can convey just as much information as the words being spoken. Incorporating emotion into machine-generated speech saves confusion and misunderstanding. Unfortunately, the generation of emotional speech is still in its early days. Therefore, we also set forth to propose a solution for introducing emotion into generated speech. [6],[7]
The contributions of this paper are twofold

- Trained classifiers that process conversational words and COVID-19-related words in SgSL
- Emotional Embedding of conversational speech

1.1 AIMS AND OBJECTIVES

This study aims to investigate the feasibility of a deep learning model that can synthesise Emotion Embedded Speech from Singapore Sign Language. The criteria of this project are the accuracy of the sign language processor and speech synthesiser (speech clarity, presence of emotion). The constraints include no existing datasets for SgSL and small datasets for emotional speech.

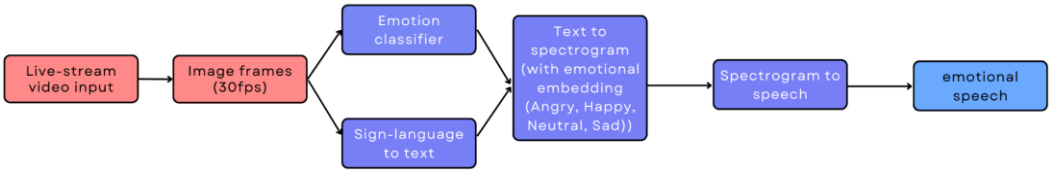


Fig.1 Pipeline

2 LITERATURE REVIEW

This section gives a brief overview into sign language processing and emotional speech synthesis as well as existing works, models and datasets catered to them.

2.1 ALTERNATIVES FOR SIGN LANGUAGE PROCESSING

Existing well-known sign language processors make use of devices that are fitted with sensors to track hand movements and convert them to speech. However, they are bulky and uncomfortable to wear. The Wearable-tech glove (Fig.2) developed by the University of California, Los Angeles (UCLA) in 2020 recognised 660 signs including numbers and alphabets. However, the glove does not consider the position of the signed hand gesture in relation to other body parts, which results in different meanings of signs. Gloves are also intrusive and inconvenient to have to wear for long time periods. Another model, DeepASL makes use of infrared light to convert words and sentences of sign language to speech with an accuracy of 94%. It also received criticism of the intrusiveness of their model and disregard to facial expressions which are a key component in conveying a message. [8],[5]

Fig.2 Wearable-tech glove by UCLA

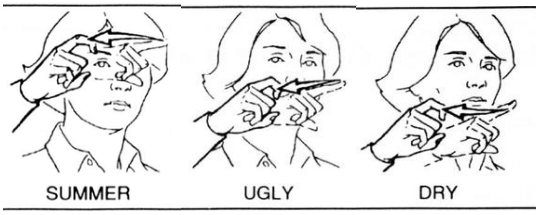


Fig.3 Signed Words performed at different locations

Fig.4 DeepASL's real-world proposal

2.2 FRAMEWORK FOR SGSL PROCESSING - LSTM AND MEDIAPIPE

Long short-term memory (LSTM) is a type of recurrent neural network (RNN) that processes specific sequences of data points such as speech or videos. The LSTM unit consists of 3 gates which are 'input gate', 'output gate' and 'forget gate' and a cell. The cell retains values over time while the gates regulate the flow of information in and out of the cell.Hence, these models are the predominant choice for action detection and recognition as well as other apt uses.

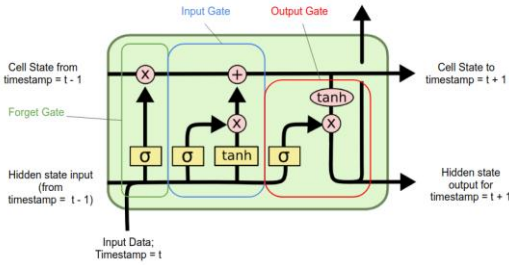


Fig.5 LSTM Cell

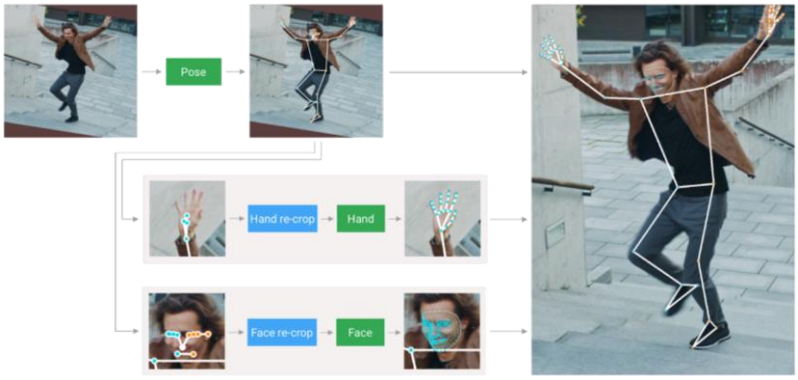


Fig.6 MediapipeHolistic Pipeline

	No. of landmarks
Face Mesh	468
Left Hand	21
Right Hand	21
Pose	33

Total	543

Fig.7 Keypoint Distribution

Using BlazePose's pose detector and subsequent keypoint model, MediaPipe Holistic calculates the human pose. It then creates three regions of interest (ROI) crops for each hand (2x) and the face using the inferred pose key points, and uses a re-crop model to enhance the ROI, as seen in Fig.6 . The pipeline then applies task-specific face and hand models to these ROIs, crops the full-resolution input frame, and estimates the appropriate key points. All key points are combined with the pose model to get the full set of 543 key points.

2.3 EMOTIONAL SPEECH SYNTHESIS

Emotion is extremely crucial in human interactions, which often takes place through speech. Due to the limited number of existing emotional speech datasets, models trained with emotion are weak and noisy. In the making of our end-to-end text-to-speech model, the Tacotron approach was used.

2.4 PAST WORKS

Tacotron 2 is a neural network framework for end-to-end text to speech synthesis. It comprises a (1) recurrent sequence-to-sequence feature prediction network that maps character embeddings to mel-scale spectrograms, and (2) a modified WaveNet model (the vocoder) which synthesises time-domain waveforms from the spectrograms, which are conditioned on the predicted mel spectrogram frames. The Tacotron 2 model is more robust than existing speech synthesisers and can be trained from scratch with random initialization. [12]

Tacotron achieves a 3.82 subjective 5-scale mean opinion score on US English, outperforming a production parametric system in terms of naturalness. In addition, since Tacotron generates speech at the frame level, it is faster than sample-level autoregressive methods.

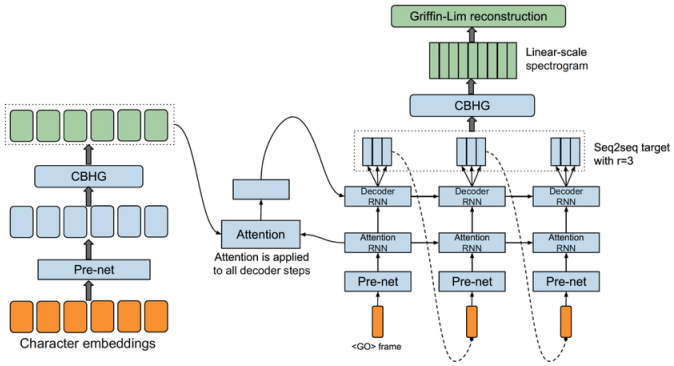


Fig.8 Tacotron pipeline

2.5 EMOTIONAL SPEECH DATASETS

An emotional speech dataset experimented with is RAVDESS (The Ryerson Audio-Visual Database of Emotional Speech and Song). It contains 7356 audio and video files of 24 professional actors vocalising two lexically-matched statements in a neutral North American accent. The database includes calm, happy, sad, angry, fearful, surprised, and disgusted speech. Each emotion is produced at two levels of intensity (normal, strong), with an additional neutral expression. [13]

3 METHODOLOGY

This section will discuss the 3 models, as well as improvements made to them, that accumulate into a final product that converts Singapore Sign Language to emotional speech. The sign language processor

and emotion classifier will extract text and emotion from a live stream, which will be fed as input for the emotional speech synthesiser.

3.1 SGSL PROCESSOR

This SGSL processor was trained in Google Colab using TensorFlow v2.9, numpy 1.21.4 and openCV. The pipeline and model architecture will be elaborated on together with mass dataset collection.

3.3.1 PIPELINE

The MediaPipe Holistic Model was used to identify and track the movement of key points of an individual's face, hands, and joints across a duration. Fig. 9 shows the position of key points across the various body features respectively. Through this, facial expressions and the relative positioning of the hands in relation to the different body parts can be captured.

Fig.9 keypoint detection

```
# Set mediapipe model
sequences, labels = [], []
with mp_holistic.Holistic(min_detection_confidence=0.8, min_tracking_confidence=0.8) as holistic:
    # NEW LOOP
    # Loop through actions
    for action in actions:
        # Loop through sequences aka videos
        dir_name = os.path.join('/content/drive/MyDrive/wasl/videos', action)
        for filename in os.listdir(dir_name):
          video_name = os.path.join(dir_name, filename)
          res_points = get_Data(video_name, action)
          sequences.append(res_points)
          labels.append(label_map[action])
```

Fig.10 For Loop that stores keypoint information

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 (None, 30, 64)            442112

 lstm_1 (LSTM)               (None, 30, 128)           98816

 lstm_2 (LSTM)               (None, 64)                49408

 dense (Dense)               (None, 64)                4160

 dense_1 (Dense)             (None, 32)                2080

 dense_2 (Dense)             (None, 3)                 99

=================================================================
Total params: 596,675
Trainable params: 596,675
Non-trainable params: 0
_________________________________________________________________
```

Fig. 11 Model Architecture

A total of 15 classes of common conversational words and phrases were chosen as signs: hello, how are you, bye, sorry, thank you, please, see you later, yes, no, help, fine, more, I, you and love. They were a mix of static and dynamic signs, as well as single and double hand gestures. The gestures used were based on youtube tutorial videos on sign language called '25 ASL Signs You Need to Know | ASL Basics | American Sign Language for Beginners' and 'LEARN SgSL - SINGAPORE SIGN LANGUAGE 101: Greetings', the latter recorded by ExtraOrdinary Horizons, a local Deaf community.

1-second video clips of 30 frames for each class are the inputs of the pipeline. The x,y and z coordinates of each landmark are recorded and stored across a series of 30 frames in the form of an array, as seen in Fig. 10. These arrays are then stored under the relevant class for training. A 90/10 split of the dataset is used for training and testing. The model's architecture consists of 3 LSTM blocks and 3 Dense Layers, including relu and softmax used as activation functions. The model is then trained on 1000 epochs, with Adam used as an optimization algorithm. A multilabel confusion matrix is then used to evaluate and visualise the performance of the model, through which an overall accuracy score is obtained.

3.3.2 MASS DATASET COLLECTIONS

A mass original dataset for SgSL conversational phrases was created by first recording a mirrored tutorial video of 15 signs. The tutorial video was then sent to over 55 unique individuals who then recorded themselves performing the required signs. A variation of lighting, masked vs. unmasked actors and angle of recording allowed the data to be more representative of real-world situations. These videos were manually split into 1-second clips for each sign and sorted into their respective classes. As a secondary level of data preprocessing, the clips were manually cropped to a square frame and centralised with their head to the torso in all frames using an application called Canva, as seen in Fig. 12.

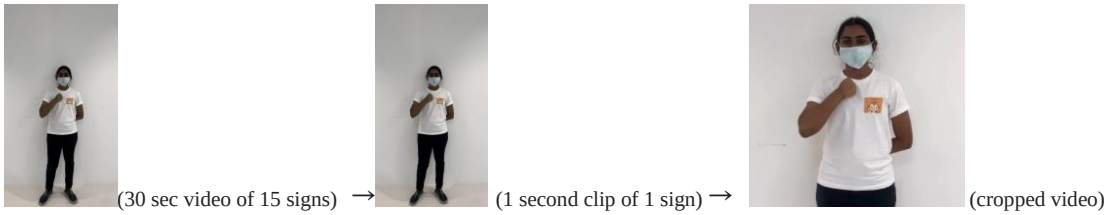


Fig.12 Canva Preprocessing

Fig. 13 (appendix) shows the results of the initial run for every new class added. The F1 scores were tabulated using `multilabel_confusion_matrix`. There was a significant decrease in accuracy for each additional class added. This could have been due to the increased noise due to great variations in gestures done by each individual. Additionally, when some key points on the hands or elbows went out of frame for a short duration, their coordinates were replaced with (0,0,0). This resulted in the model being unable to learn and differentiate between various signs due to the inconsistent pattern in the movement of the key points.

As such a second dataset, Dataset B, was recorded using a built-in camera of the same laptop for standardisation. This way, the camera resolution and frame rate were kept constant. Individuals were filmed from a distance such that their head to the torso was covered and all landmarks on the hands were ensured to be in the frame throughout the recording. A variation of lighting and masked vs unmasked individuals were used for filming. Fewer individuals were used for filming to reduce noise due to incorrect gestures, another issue that was prevalent in Dataset A, that significantly impacted its accuracy. Dataset B performed much better for every additional class added due to the better clarity of actions used to train the LSTM model as well as the greater videos per class due to repetitions carried out in different environments. As seen in Fig.13, the accuracy of the model fared much better, with a much lower loss in accuracy per class.

Another SgSL dataset was created in the same way for words specific to COVID-19. Following a reference video by the Singapore Association of the Deaf, 10 most common words were chosen and recorded. These signs were more complex with more parts to it, testing the calibre of the model architecture in learning harder signs. These signs would also be more predominantly used in the new normal and post pandemic world, making it more relevant to be trained by the SgSL processor.

3.2 EMOTION CLASSIFIER

A trained VGGNet model was used to first identify the emotion displayed by the signer. The classes were then adjusted to identify 4 main emotions: happy, sad, angry and neutral. Fig. 14 and Fig.15 in the appendix show the model architecture. It consists of 8 blocks of Conv2D that include Batch Normalization, MaxPool2D and dropout for every 2 Conv2D blocks. There are 3 dense layers with Adam used as an optimization algorithm. Using the mediapipe face mesh (Fig.16 in appendix), a bounding box is drawn around the face and is fed as input into the model.

3.3 EMOTIONAL SPEECH SYNTHESIS

3.3.1 DATASET CREATION

An original dataset of English emotional speech files was created in these emotions: angry, happy and sad. The database has a total of 3012 audio files authored by a single speaker, largely tailored to conversational speech. These audio tapes were recorded with an external microphone with minimal ambient noise to preserve clarity. Variation in expression was attempted while recording the same words for each emotion. They were then processed and trimmed in Audacity, an audio processing

interface, and sorted into folders by emotion. The captions for the audio files were formatted in a similar manner to that LJ-Speech.

3.3.2 MODEL ARCHITECTURE

A Tacotron2 model, composed of a (1) sequence-to-sequence (seq2seq) and a (2) vocoder. (1) This seq2seq model is made of an encoder and decoder. The encoder, made of a Prenet and a CBHG block, extracts the sequential representation of text. The decoder, consisting of a pre-net, attention RNN and decoder RNN, is an attention mechanism to generate mel spectrograms. (2)The vocoder utilises Griffin Lim algorithm, which reconstructs a signal from its magnitude spectrogram, which is obtained from the STFT (Short-time Fourier transform) of the signal.

To train Tacotron2, a large dataset of text and corresponding audio recordings is required. Our pipeline includes a preprocessor which converts wav files into spectrograms and tokenizes the text. During training, the model will learn to map text input to spectrogram output, and it will use an autoregressive decoder to generate an audio waveform from the spectrogram.
To create an optimal end-to-end text to speech model, we ran through several experiments. The hyperparameters of the final model can be found in Fig 19 (Appendix).

3.3.3 INITIAL RUN

A vanilla Tacotron2 model was pre-trained on LJ-Speech, producing promising results. Due to LJ-Speech being a single-speaker dataset and its vast lexicon coverage, natural sounding speech was generated.

3.3.4 EXPERIMENTS

1. Fine-tuning pre-trained Tacotron2 on RAVDESS

The Tacotron2 model was fine-tuned with angry speech from RAVDESS (an emotional speech dataset as mentioned in 2.7). The RAVDESS dataset was segmented by the various emotions into folders and the different emotional models were trained separately. In training the dataset, 500 epochs and a high learning rate α=1e-2 was used so the model could learn faster, considering that RAVDESS is a small dataset. However, the database covers only two statements: "Kids are talking by the door" and "Dogs are sitting by the door". So, the trained model generated unintelligible results for phrases not in the dataset. Even as the input text matched the captions for the audio files, the trained model generated nonsensical speech, with fragments of the expected speech. It was suspected that the poor results are caused by the multi-speaker approach – there is poor alignment in utterances amongst speakers, as different speakers have different speaking speeds. The loss graph for this model can be seen in Fig 14 (appendix).

2. Fine-tuning pre-trained Tacotron2 on custom dataset

The Tacotron2 model was fine-tuned on our original dataset as mentioned in 3.3.1. In view of the conversational phrases our sign-to-text model is able to detect, Tacotron2 was trained on conversational English instead. In doing so, we utilised SGD (Stochastic Gradient Descent) as our optimizer. SGD optimizers are known to generalise better when the data is very sparse. Our trained model diverged possibly due to the high learning rate of 1e-3 (lower than in 3.3.4.1). The generated text was unintelligible, even when it was part of the dataset. The loss graph for this model can be seen in Fig 14 (appendix).

3. Changing optimizer to Adam and decreasing learning rates

From 3.3.4.2, we changed the optimizer to Adam and used a smaller learning rate α = 5e-4,correspondingly decreased batch size to 78 and increased the epochs to 700. Adam includes an adaptive learning rate for each parameter, allowing our model to converge faster. The smaller learning rates ensures that the weights from pretraining are not overwritten, preventing the model from

diverging from the optimal output. The trained model produces clear and intelligible speech with recognisable emotion. The loss graph for this model can be seen in Fig 14 (appendix).

RESULTS AND DISCUSSION

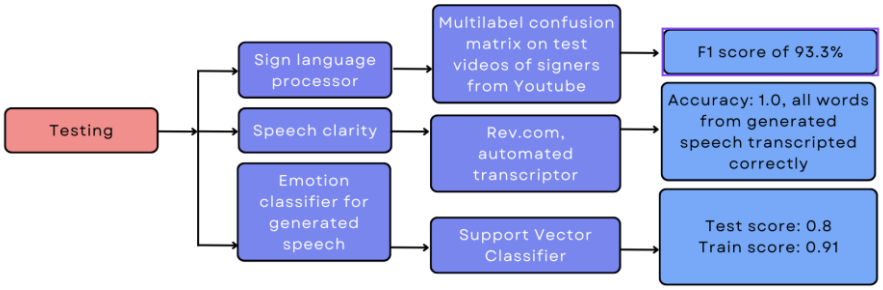


Fig.21 Testing procedures

The final model was a combination of the two datasets with further data cleansing to ensure only accurate representations of the sign were allowed to train. Data augmentation was also carried out to include random text, random noise and random rotation to better train the model and prevent overfitting. The final model had an accuracy of 93.3%.

After training the separate models, they were loaded in an inference notebook on Google Colab. Input text was converted to spectrograms and speech using our model (Fig. 15 in Appendix).

To verify the intelligibility of the speech generated, the generated audio files were saved and uploaded to Rev.com, where they were passed through an automated transcriptor. The results are seen in Fig. 16 (Appendix). This gives us a 1.00 accuracy for the test input phrases.

A Support Vector Classifier (SVC) model was trained on our dataset to test for emotion in our generated speech. Our custom dataset was added to the training datapath and our generated audio files was added to the testing datapath, with 5 audio files for each emotion (happy, angry, sad). We obtained Test score: 0.8 and Train score: 0.915657. The results are seen in Fig 18. (Appendix).

CONCLUSION

In future, we plan to do the following works:

1) Expand the dataset to cover more phrases in Singapore Sign Language
2) Explore Natural Language Processing for sentence-level translation of Sign Language to Spoken text due to differences in grammar and syntax of the 2 languages
3) Create Speech Recognition and Sign Language Synthesiser models to facilitate 2 - way communication between parties
4) Mass testing through stronger alliance with Singapore Association of the Deaf

Ultimately, like any language, the translation of sign language is complex, multifaceted and a truly effective model remains yet to be seen. This project has only attempted a word level translation of Sign Language although the syntax for Sign Language is not the same as spoken English. Additionally, the model may not be able to match the speed of the signs performed by a signer, troubling a signer to have to slow down their pace or creating avenues of miscommunication. Despite these limitations, there is still potential for growth in alleviating some of the struggles of the deaf community. Our model can be used as an app (Fig.17 in appendix) to facilitate day-to-day conversations between the signing and non-signing community as well as teleconferencing in an increasingly digital world (Fig. 17. To conclude, this project has achieved its goals of (indicators + data). We hope that this research project has taken a more inclusive approach to bridge communication between the signing and non-signing community, bringing Singapore one step closer to caring for and empowering its local deaf community.

RO031
Sign2Speak! Synthesising Emotional Speech from Singapore Sign Language through Deep Learning
Mirdhini Shri Rajaram, Ong Yu Xi

APPENDIX

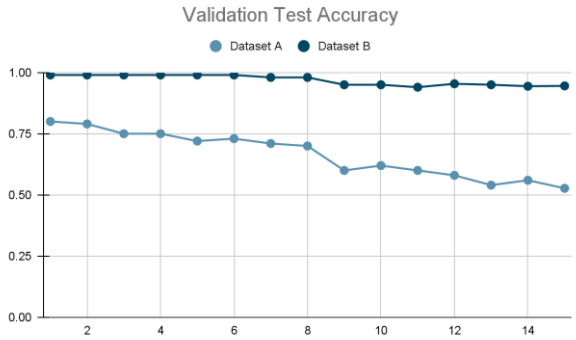
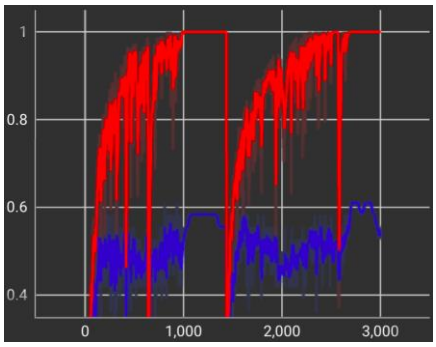
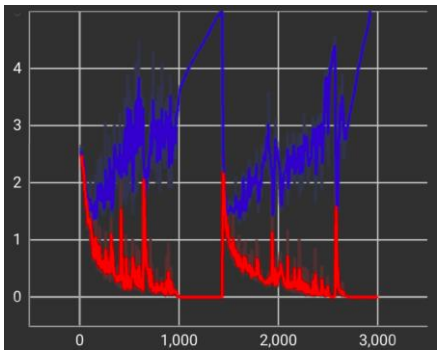
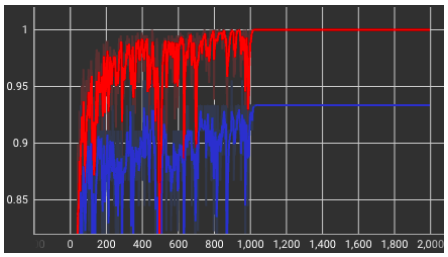
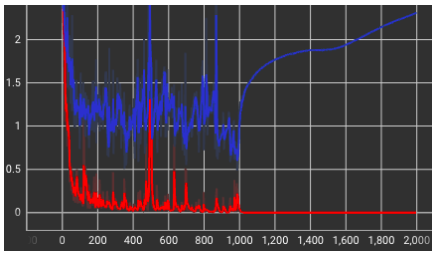


Fig.13 Accuracy for every new class (sign) added

<table>
<tr><td colspan="3">Dataset A Results</td></tr>
<tr><td> 
</td><td> 
</td><td rowspan="2">Low validation accuracy and high validation loss due to increased noise from incorrect signs performed in mass dataset.</td></tr>
<tr><td>Accuracy</td><td>Loss</td></tr>
<tr><td colspan="3">Dataset B Results</td></tr>
<tr><td> 
</td><td> 
</td><td rowspan="2">Validation accuracy has increased to nearly twice of before and has converged. However, increasing loss suggests that error rate of classifying validation videos is high, suggesting that model is overfitted.</td></tr>
<tr><td>Accuracy</td><td>Loss</td></tr>
<tr><td colspan="3">Final Dataset Results (after data augmentation)</td></tr>
</table>

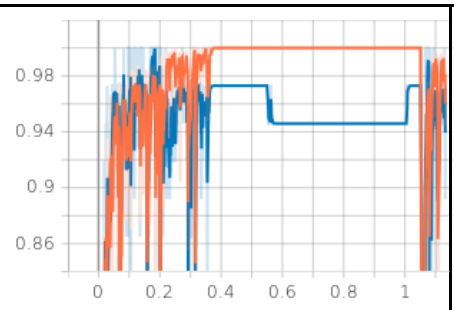
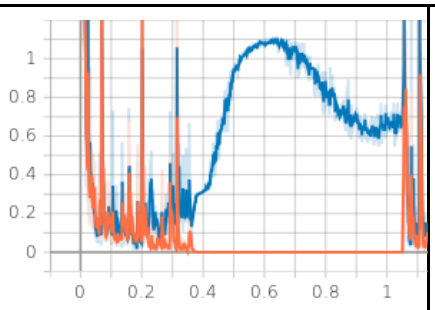
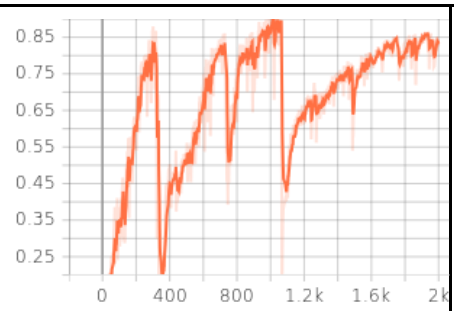
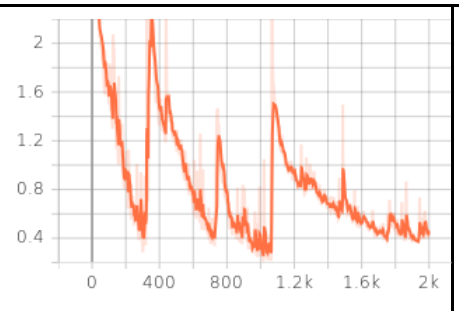
		Lower validation loss suggests that the overall model has become more precise with a lower error rate.
Accuracy	Loss	
Covid-19 Dataset		
		
Accuracy (TBC)	Loss (TBC)	

Figure for video augmentation

Multilabel Confusion Matrix

Fig.14 VGGNet Model Architecture

Layer (type)	Output Shape	Param #
rescaling (Rescaling)	(None, 48, 48, 1)	0
conv2d (Conv2D)	(None, 46, 46, 64)	640
batch_normalization (BatchNormalization)	(None, 46, 46, 64)	256
conv2d_1 (Conv2D)	(None, 46, 46, 64)	36928
batch_normalization_1 (BatchNormalization)	(None, 46, 46, 64)	256
max_pooling2d (MaxPooling2D)	(None, 23, 23, 64)	0
dropout (Dropout)	(None, 23, 23, 64)	0
conv2d_2 (Conv2D)	(None, 23, 23, 128)	73856
batch_normalization_2 (BatchNormalization)	(None, 23, 23, 128)	512
conv2d_3 (Conv2D)	(None, 23, 23, 128)	147584
batch_normalization_3 (BatchNormalization)	(None, 23, 23, 128)	512
max_pooling2d_1 (MaxPooling2D)	(None, 11, 11, 128)	0
dropout_1 (Dropout)	(None, 11, 11, 128)	0
conv2d_4 (Conv2D)	(None, 11, 11, 256)	295168
batch_normalization_4 (BatchNormalization)	(None, 11, 11, 256)	1024
conv2d_5 (Conv2D)	(None, 11, 11, 256)	590080
batch_normalization_5 (BatchNormalization)	(None, 11, 11, 256)	1024
max_pooling2d_2 (MaxPooling2D)	(None, 5, 5, 256)	0
dropout_2 (Dropout)	(None, 5, 5, 256)	0
conv2d_6 (Conv2D)	(None, 5, 5, 512)	1180160
batch_normalization_6 (BatchNormalization)	(None, 5, 5, 512)	2048
conv2d_7 (Conv2D)	(None, 5, 5, 512)	2359808
batch_normalization_7 (BatchNormalization)	(None, 5, 5, 512)	2048
max_pooling2d_3 (MaxPooling2D)	(None, 2, 2, 512)	0
dropout_3 (Dropout)	(None, 2, 2, 512)	0
flatten (Flatten)	(None, 2048)	0
dense_7 (Dense)	(None, 1024)	2098176
dropout_4 (Dropout)	(None, 1024)	0
dense_8 (Dense)	(None, 256)	262400
dense_9 (Dense)	(None, 7)	1799

Total params: 7,054,279
Trainable params: 7,050,439
Non-trainable params: 3,840

Fig.15 VGGNet Model Parameters

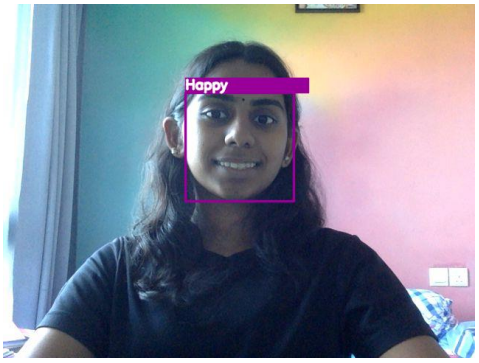
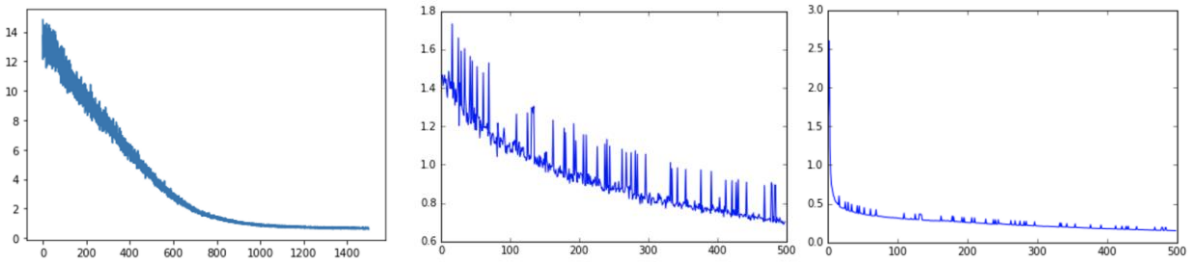


Fig.16 Bounding Box on Face



1. Loss (y) against iterations (x). Model approaches convergence.

2. Loss (y) against epochs (x). Model is unable to converge.

3. Loss (y) against epochs (x). Model is able to converge

Fig.17 Graphs for experiments in 3.3.4

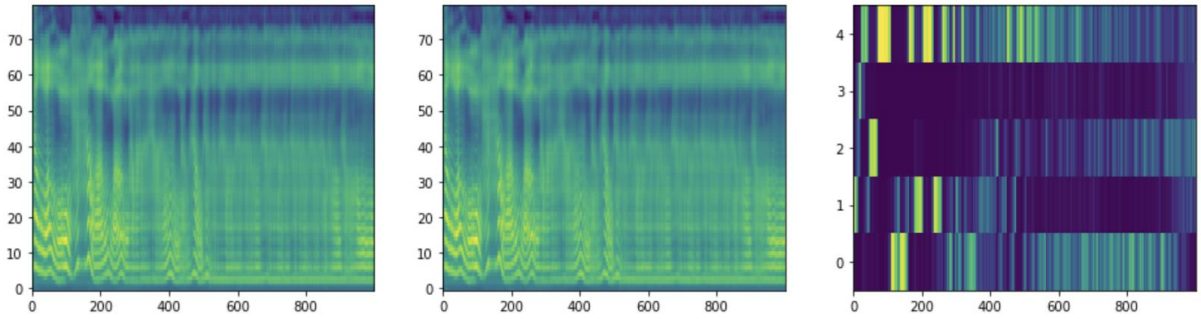


Fig.15 (From left to right: Spectrogram of mel_outputs, mel_outputs_postnet, and alignments for input “hello”

Input label (angry model)	Automated transcription	Input label (happy model)	Automated transcription	Input label (sad model)	Automated transcription
hello	Hello?	hello	Hello,	hello	hello.
how are you	How are you?	how are you	How are you?	how are you	How are you?
sorry	sorry,	sorry	sorry,	sorry	sorry.
please help me	please help me.	please help me	please help me.	please help me	please help me.
yes	Yes.	yes	Yes,	yes	yes
no	No,	no	No.	no	No.
thank you	Thank you.	thank you	Thank you.	thank you	Thank you.

Fig.16 Testing results for generated speech

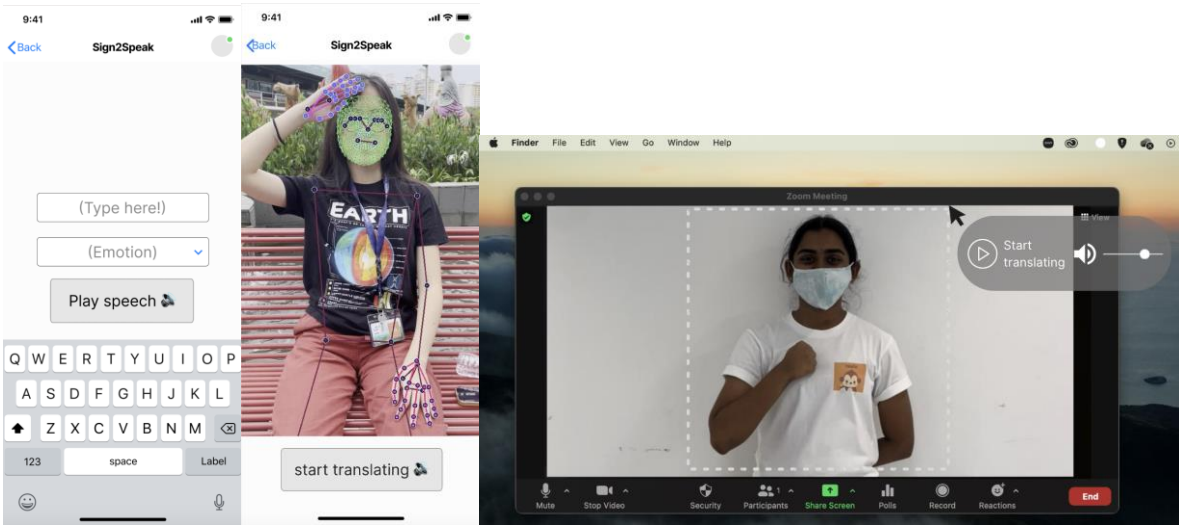


Fig.17 Sign2Speak App Interface (Mobile and Desktop versions)

```
print("Test score:", rec.test_score())
# check the train accuracy for that model
print("Train score:", rec.train_score())
```

```
['sad' 'angry' 'happy' 'sad' 'angry' 'angry' 'sad' 'sad' 'angry' 'happy'
 'happy' 'happy' 'happy' 'sad' 'angry'] real
['happy' 'angry' 'angry' 'sad' 'angry' 'angry' 'sad' 'sad' 'angry' 'happy'
 'happy' 'happy' 'happy' 'happy' 'angry'] pred

Test score: 0.8
Train score: 0.9156577885391445
```

Fig.18 Testing results (emotion in generated speech)

```
hparams = {
    'epochs':700,
    'iters_per_checkpoint':1000,
    'seed':1234,
    'dynamic_loss_scaling':True,
    'fp16_run':False,
    'distributed_run':False,
    'dist_backend':"nccl",
    'dist_url':"tcp://localhost:54321",
    'cudnn_enabled':True,
    'cudnn_benchmark':False,
    'ignore_layers':['embedding.weight'],

    ################################
    # Data Parameters             #
    ################################
    'load_mel_from_disk':False,
    'training_files':'/content/drive/MyDrive/tacotron2/filelists/angry_audio_text_train_filelist.txt',
    'text_cleaners':['english_cleaners'],

    ################################
    # Audio Parameters             #
    ################################
    'max_wav_value':32768.0,
    #'sampling_rate':22050,
    'sampling_rate':48000,
    'filter_length':1024,
    'hop_length':256,
    'win_length':1024,
    'n_mel_channels':80,
    'mel_fmin':0.0,
    'mel_fmax':8000.0,
```

```
# Location Layer parameters
'attention_location_n_filters':32,
'attention_location_kernel_size':31,

# Mel-post processing network parameters
'postnet_embedding_dim':512,
'postnet_kernel_size':5,
'postnet_n_convolutions':5,

################################
# Optimization Hyperparameters #
################################
'use_saved_learning_rate':False,
'learning_rate':5e-4,
'weight_decay':1e-7,
'grad_clip_thresh':1.0,
'batch_size':78,
'mask_padding':True  # set model's padded outputs to padded values
```

Fig.19 Tacotron model hyperparameters

```
'hop_length':256,
'win_length':1024,
'n_mel_channels':80,
'mel_fmin':0.0,
'mel_fmax':8000.0,

################################
# Model Parameters             #
################################
'n_symbols':len(symbols),
'symbols_embedding_dim':512,

# Encoder parameters
'encoder_kernel_size':5,
'encoder_n_convolutions':3,
'encoder_embedding_dim':512,

# Decoder parameters
'n_frames_per_step':1,  # currently only 1 is supported
'decoder_rnn_dim':1024,
'prenet_dim':256,
'max_decoder_steps':1000,
'gate_threshold':0.5,
'p_attention_dropout':0.1,
'p_decoder_dropout':0.1,

# Attention parameters
'attention_rnn_dim':1024,
'attention_dim':128,
```

REFERENCES

1 Cai, C. (2022) *An ESCAP Guide towards Legal Recognition of Sign Languages in Asia and the Pacific*. https://www.unescap.org/kp/2022/sign-language-what-it-escap-guide-towards-legal-recognition-sign-languages-asia-and-pacific

2 (2021) Singapore Sign Language. *Ethnologue*, www.ethnologue.com/language/sls/25

3 SaDeaf (2022) FAQ on Sign Language https://www.Sadeaf.org.sg

4 Yin, K. (2021). Rule of Thumb: How big should your emergency fund be? https://www.thebalance.com/is-your-emergency-fund-too-big-4142617

5 Matchar, E. (2019) Sign Language Translating Devices Are Cool. But Are They Useful? https://www.smithsonianmag.com/innovation/sign-language-translators-are-cool-but-are-they-useful-180971535/

6 John Hopkins University. (2021). Emotional Speech https://engineering.jhu.edu/nsa/research/emotional-speech/#:~:text=Emotion%20is%20the%20cornerstone%20of,vocal%20inflections%20known%20as%20prosody.

7 Shen, J. (2017). Tacotron 2: Generating Human-like Speech from Text. https://ai.googleblog.com/2017/12/tacotron-2-generating-human-like-speech.html

8 Chin, M. (2020). Wearable-tech glove translates sign language into speech in real time. https://newsroom.ucla.edu/releases/glove-translates-sign-language-to-speech

12 Wang, Y. (2017). Tacotron: Towards End-to-End Speech Synthesis https://arxiv.org/abs/1703.10135

13 Livingstone, S. (2018) The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5955500/#:~:text=The%20RAVDESS%20is%20a%20validated,a%20neutral%20North%20American%20accent

14 University of Essex. (2022). Mask mandates saw more than 90% of deaf people struggle

https://www.essex.ac.uk/news/2022/10/03/covid-masks-impact-on-deaf-communities-revealed#:~:text=Mask%20mandates%20saw%20more%20than%2090%25%20of%20deaf%20people%20struggle&text=Mandatory%20mask%20wearing%20saw%20more,due%20to%20the%20face%20coverings.

Fig. 2 Chin M.(2020) Wearable-Tech Glove Translates Sign Language into Speech in Real Time https://newsroom.ucla.edu/releases/glove-translates-sign-language-to-speech

Fig.3 Fekete, E. (2017) Embodiment, linguistics, space: American Sign Language meets geography https://www.researchgate.net/publication/315983047_Embodiment_linguistics_space_American_Sign_Language_meets_geography

Fig. 4 Fang, B. (2018) DeepASL: Enabling Ubiquitous and Non-Intrusive Word and Sentence-Level Sign Language Translation https://arxiv.org/abs/1802.07584

Fig. 5 T, R. (2020) LSTMs Explained: A Complete, Technically Accurate, Conceptual Guide with Keras https://medium.com/analytics-vidhya/lstms-explained-a-complete-technically-accurate-conceptual-guide-with-keras-2a650327e8f2

Fig. 6 Grishchenko, I. (2020) MediaPipe Holistic — Simultaneous Face, Hand and Pose Prediction, on Device https://ai.googleblog.com/2020/12/mediapipe-holistic-simultaneous-face.html

Fig.8 Wang Y. (2017) Tacotron: Towards End-to End Speech Synthesis https://arxiv.org/abs/1703.10135

RO031
Sign2Speak! Synthesising Emotional Speech from Singapore Sign Language through Deep Learning
Mirdhini Shri Rajaram, Ong Yu Xi