IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Lip Reading Sentences Using Deep Learning with Only Visual Cues

SOUHEIL FENGHOUR[1], DAQING CHEN[1], KUN GUO[2], AND PERRY XIAO[1]
[1]School of Engineering, London South Bank University, London, UK, SE1 0AA
[2]Xi'an VANXUM Electronics Technology Co., Ltd., China

Corresponding author: Souheil Fenghour (e-mail: fenghous@lsbu.ac.uk).

This study was supported under a joint scholarship by Chinasoft International Ltd., and London South Bank University.

ABSTRACT In this paper, a neural network-based lip reading system is proposed. The system is lexicon-free and uses purely visual cues. With only a limited number of visemes as classes to recognise, the system is designed to lip read sentences covering a wide range of vocabulary and to recognise words that may not be included in system training. The system has been testified on the challenging BBC Lip Reading Sentences 2 (LRS2) benchmark dataset. Compared with the state-of-the-art works in lip reading sentences, the system has achieved a significantly improved performance with 15% lower word error rate. In addition, experiments with videos of varying illumination have shown that the proposed model has a good robustness to varying levels of lighting. The main contributions of this paper are: 1) The classification of visemes in continuous speech using a specially designed transformer with a unique topology; 2) The use of visemes as a classification schema for lip reading sentences; and 3) The conversion of visemes to words using perplexity analysis. All the contributions serve to enhance the accuracy of lip reading sentences. The paper also provides an essential survey of the research area.

INDEX TERMS deep learning, lip reading, neural networks, perplexity analysis, speech recognition.

I. INTRODUCTION

THE task of automated lip reading has attracted a lot of research attention in recent years and many breakthroughs have been made in the area with a variety of machine learning-based approaches having been implemented [1] [2]. Automated lip reading can be done both with and without the assistance of audio [3] and when performed without the presence of audio, it is often referred to as visual speech recognition [4].

The most recent approaches to automated lip reading are deep learning-based and they largely focus on decoding long speech segments in the form of words and sentences using either words or ASCII characters as the classes to recognise [5] [6] [7] [8] [9] [10]. Lip reading systems that are designed to classify words often use individual words as the classification schema where every word is treated as a class. In recent years, very good accuracies have been achieved for word-based classification on some of the most challenging audio-visual datasets for words, such as LRW [7] and LRW-1000 [48].

Contrastingly, however, lip reading sentences have not succeeded in attaining accuracies as good as word-based approaches. It still remains an ongoing challenging task to automatically lip reading people uttering sentences which cover a wide range of vocabulary and contain words that may not have appeared in the training phase while using the fewest classes possible. The main obstacles to lip reading sentences are:

- Lip reading systems that use words or ASCII characters as classes can only predict words that the systems have been trained to predict because in the case of using words as a class, the word needs to be encoded as a class and presented in the training phase; while in the case of ASCII characters, the prediction of words is based on combinations of characters having been presented in the training phase as patterns.
- The models must be trained to cover a wide range of vocabulary which requires a significant number of parameters in the models to be optimised and a significant volume of training data to be used.
- They often require curriculum learning-based strategies [28] [29] which involve further pre-processing, whereby the videos of individuals speaking in the training data have to be clipped so that the models can be trained on single word examples initially, with the length of the sentences being gradually incremented.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3040906, IEEE Access

This paper focuses on improving the accuracy of lip reading sentences and this is achieved by using visemes as a very limited number of classes for classification, a specially designed deep learning model with its own network topology for classifying visemes, and a conversion of recognised visemes to possible words using perplexity analysis.

Using visemes for lip reading sentences has some unique advantages. The use of visemes as classes in comparison to the use of either words or ASCII characters as classes requires an overall smaller number of classes which alleviates bottleneck in the computation. In addition, using visemes does not require pre-trained lexicons, meaning that a viseme-based lip reading system can be used to classify words that have not presented in the training phase, and they can be generalised to different languages because many different languages share the same visemes.

On the other hand, there are some specific issues to be considered when designing a viseme-based lip reading system for sentences. The general classification performance for individual segmented visemes has been less satisfactory in comparison to the classification of words due to the fact that visemes tend to have a shorter duration than words. This results in there being less temporal information available to distinguish between different classes, as well as there being more visual ambiguity when it comes to class recognition [25]. One possible way to address this problem is to significantly increase the training data available to enhance the system's ability to distinguish between classes, and this is why a high volume of training videos have been utilised. Moreover, there is a direct conversion of recognised ASCII characters to possible words in a one-to-one mapping relationship, whereas this one-to-one mapping relationship does not exist when using visemes, because one set of visemes can map to multiple different sounds or phonemes. This also means that once visemes have been classified, there is still the need to perform a viseme-to-word conversion. This approach also helps to distinguish between homopheme words or words that look the same when spoken but sound different [11], a phenomenon that exists because of the one-to-many mapping relationship between visemes and phonemes.

The proposed automated lip reading system contains a component to classify spoken visemes from people speaking in silent videos, and a component to perform viseme-to-word conversions using perplexity analysis [12]. The proposed model also has a good robustness to varying levels of lighting.

The rest of the paper is organised as follows: First in Section II, the different classification schema for automated lip reading are discussed along with their advantages and limitations. Then in Section III, details of all the components that make up the whole lip reading system including pre-processing, visual feature extraction, viseme classification and word detection are given. In Section IV, the classification results for the overall lip reading system are discussed and compared followed by concluding remarks given in Section V along with suggestions for further research.

II. LITERATURE REVIEW

Automated lip reading systems initially focused on classifying isolated speech segments in the form of digits and letters [13] [14] [15] [16] [17], and then eventually moved on to longer speech segments in the form of words. The success of automated lip reading was previously constrained by the available training data, as initially, the only audio-visual datasets available were those with isolated speech segments, i.e., digits, alphabet and words [18] [19] [20]. Subsequently every speech segment was treated as a class to recognise.

Thanks in part to the availability of larger audio-visual datasets with continuous speech, later lip reading systems have focused on classifying entire sentences utilising a wider range of vocabulary and so have opted for ASCII-based class systems [5] [6] [7] [8] [9] [10]. Sentences are spelt using ASCII characters as opposed to including a class for every single word, which allows for the use of fewer classes and avoids the creation of computational bottleneck [31]. ASCII characters also allow for the modelling of natural language due to the conditional probability relationships that exist between ASCII characters making it easier to predict characters and words.

Very good accuracies have been attained in some of the most recent neural network-based lip reading systems that are trained to classify individual words on word-based lip reading datasets like LRW [7] and LRW-1000 [48]. LRW is a very taxing dataset since it consists of more than 1000 speakers with large variations in head pose and illumination. LRW-1000 is an even more tricky Mandarin lip reading dataset, due to its large variations in scale, resolution and background clutter.

Notable performances have been recorded for lip reading systems that predict entire sentences, such as those predicting phrases from the GRID [49] and OuluVS [50] datasets. However, sentences in datasets like GRID and OuluVS are simple, repetitive and follow standard sequences unlike those contained within the LRS2 corpus which are more random and varied. A summary of the most recent state-of-the-art lip reading models and their performances is given in Table 1.

Other alternative classification schemas for neural network-based lip reading include phonemes which have been used in audio and acoustic speech recognition systems [21]. Assael et al. [10] used a neural network architecture consisting of a spatial-temporal convolutional neural network(CNN) and a Long-Short Term Memory Network (LSTM) to classify sequences of phonemes from silent videos where phonemes were then mapped to words using a Finite-state transducer [31]. However, with phonemes, there is still the one-to-many mapping problem where different phonemes map to the same viseme thus producing identical lip movements.

To the best of our knowledge, there is no lip reading sentences system that has decoded entire sequences of visemes, although there has been a lot of work on classifying individual segmented visemes in the form of images or groups of image frames [23] [24] [25] [26]. If visemes are to be classified, they should be classified in the context of continuous speech in order to perform viseme classification in real-time. There is

TABLE 1: Different approaches to automated lip reading.

Year	Reference	Feature Extractor	Classifier	Database	Recognition Task	Class	Accuracy Result(%)
2017	Chung and Zisserman [51]	CNN	LSTM+attention	OuluVS2 [50]	Phrases	ASCII	91.10 [51]
2017	Chung and Zisserman [51]	CNN	LSTM+attention	MV-LRS [6]	Sentences	ASCII	43.60 [51]
2017	Chung et al. [6]	CNN	LSTM+attention	LRW [7]	Words	ASCII	76.20 [6]
2017	Chung et al. [6]	CNN	LSTM+attention	GRID [49]	Phrases	ASCII	97.00 [6]
2017	Chung et al. [6]	CNN	LSTM+attention	LRS2 [6]	Sentences	ASCII	49.80 [6]
2017	Petridis et al. [52]	Autoencoder	LSTM	OuluVS2 [50]	Phrases	ASCII	84.50 [52]
2017	Petridis et al. [53]	Autoencoder	Bi-LSTM	OuluVS2 [50]	Phrases	ASCII	91.80 [53]
2017	Petridis et al. [54]	Autoencoder	Bi-LSTM	OuluVS2 [50]	Phrases	ASCII	94.70 [54]
2017	Stafylakis and Tzimiropoulos [5]	3D-CNN+ResNet	Bi-LSTM	LRW [7]	Words	Words	83.00 [5]
2018	Afouras et al. [8]	3D-CNN+ResNet	Bi-LSTM+ Language Model	LRS2 [6]	Sentences	ASCII	37.80 [8]
2018	Afouras et al. [8]	3D-CNN+ResNet	Depthwise CNN	LRS2 [6]	Sentences	ASCII	45.00 [8]
2018	Afouras et al. [8]	3D-CNN+ResNet	Attention encoder+ Language Model	LRS2 [6]	Sentences	ASCII	50.00 [8]
2018	Fung and Mak [55]	3D-CNN	Bi-LSTM	OuluVS2 [50]	Phrases	Phrases	87.60 [55]
2018	Petridis et al. [56]	3D-CNN+ResNet	Bi-GRU	LRW [7]	Words	Words	82.00 [56]
2018	Petridis et al. [57]	Autoencoder	Bi-LSTM	AV Digits [57]	Phrases	Phrases	69.70 [57]
2018	Petridis et al. [57]	Autoencoder	Bi-LSTM	AV Digits [57]	Digits	Digits	68.00 [57]
2018	Wand et al. [58]	Feed-forward	LSTM	GRID [49]	Phrases	Words	84.70 [58]
2018	Xu et al. [59]	3D-CNN+highway	Bi-GRU+Attention	GRID [49]	Phrases	ASCII	97.10 [59]
2018	Mattos et al. [64]	CNN		GRID [49]	Visemes	Visemes	64.80 [64]
2018	Oliveira et al. [25]	CNN		GRID [49]	Visemes	Visemes	67.3 [25]
2019	Shillingford et al. [10]	3D-CNN	Bi-LSTM+ Finite-state transducer	LSVSR [10]	Sentences	Phonemes	59.10 [10]
2019	Shillingford et al. [10]	3D-CNN	Bi-LSTM+ Finite-state transducer	LRS3-TED [63]	Sentences	Phonemes	44.90 [10]
2019	Wang [60]	3D-CNN	Bi-Conv-LSTM	LRW [7]	Words	Words	83.34 [60]
2019	Wang [60]	3D-CNN	Bi-Conv-LSTM	LRW-1000 [48]	Words	Words	36.91 [60]
2020	Weng [61]	3D-CNN	Bi-LSTM	LRW [7]	Words	Words	84.11 [61]
2020	Martinez et al. [62]	3D-CNN+ResNet	Temporal CNN	LRW [7]	Words	Words	85.30 [62]
2020	Martinez et al. [62]	3D-CNN+ResNet	Temporal CNN	LRW-1000 [48]	Words	Words	41.40 [62]

one paper about an LSTM that takes visemes as an input and predicts the words that were spoken by individuals from a limited dataset with some satisfactory results [27], though the individual visemes were already known.

In addition to being treated as individual segments, visemes can also be modelled in the form of clusters like "visual words" where groups of visemes that make up a word can be segmented. Whilst approximately 50% of the words in the English language share identical viseme clusters, there are words that have unique visemes and can be classified when performing automated lip reading using solely visual information. For words that share visemes, clusters of visemes in combination would need to be analysed to determine which combination is most linguistically probable. This is the basis for the lip reading sentence system proposed in this paper based entirely on visual cues.

No official standard convention for defining precise visemes or even the precise total number of visemes exists and different approaches to viseme classification have used varying numbers of visemes as part of their conventions with different phoneme-to-viseme mappings [30] [31] [32] [33] [34] [35]. All the different conventions consist of consonant visemes, vowel visemes and one silent viseme; but Lee and Yook's mapping convention of [30] appears to be the most favoured for speech classification and it is the one that has been utilised for this paper. However, it is accepted that there are multiple phonemes that are visually identical on any given speaker [36] [37].

The different automated lip reading approaches summarised in Table 1 indicate many challenges still hindering the success of automated lip reading systems. One of these challenges is the lack of temporal information required to distinguish between segments of speech which is why some of the approaches tasked to classify shorter segments, such as visemes and digits, have not attained as good accuracies as those tasked to classify words. This problem however can be compensated for by increasing the training data available and when a small limited number of speech segments are to be classified, such as in the case of digits or visemes, the performance of such systems can be enhanced by generating as much training data as possible to train the networks.

To apply such an approach is not feasible for the case of words where the number of possible words that can be spoken is unlimited so it is necessary to use a discrete class system to cover general speech such as in the case of ASCII characters. However, the use of ASCII characters in lip reading relies on the conditional dependence relationship that exists between the characters, and ASCII symbols are not always phonetic

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3040906, IEEE Access

because of silent letters and digraphs, so to train a network to decode speech in real time requires training to have been done on an extensive range of vocabulary.

Lip reading systems tasked for predicting sentences from sentence datasets such as GRID and OuluVS have been more fruitful in terms of accuracy compared with those tasked to recognise sentences from more challenging datasets like LRS2. One of the main reasons that a dataset like LRS2 is so difficult is because it contains sentences that randomly cover a vocabulary of over 40,000 words, which is very different to the circumstances of the datasets GRID and OuluVS that contain repetitive sentences following a standard sequence, and that only cover a small range of vocabulary. Lip reading systems that use ASCII characters as classes are designed to predict words as combinations of ASCII characters and so to recognise any set of words, such words will need to have appeared in the training phase. As of present an ASCII-based lip reading systems are not be able to decode words that have not presented in training. The low accuracy of lip reading systems designed for lip reading sentences can be explained by the inability to generalize to a wide range of vocabulary whilst using a limited number of classes.

Training ASCII-based lip reading systems to generalise to a wide range of vocabulary remains an ongoing obstacle to tackle. One alternative to having a lip reading system designed for decoding speech that covers a given vocabulary range is to recognise lip movements and map them to possible words because there are distinct number of visemes that can be uttered by someone speaking. However, because of the one-to-many mapping relationship that exists between visemes and phonemes, one would still need to determine which combination of words have been uttered.

III. METHODOLOGY

Given a silent video of a talking face, the objective here is to predict the sentences being spoken by extracting their lip movements. In this Section, an overall architecture is proposed for decoding visual speech illustrated in Figure 1. The entire process consists of different stages, starting off with a Data Preprocessing stage where the region of interest is extracted from the videos using facial landmark detection to provide the input to the Visual Frontend. The components of the overall architecture include: a spatial-temporal visual frontend that inputs a sequence of images of loosely cropped lip regions, and outputs one feature vector per frame; a sequence processing module known as the viseme classifier that inputs the sequence of per-frame feature vectors and outputs a sequence of visemes, and finally a module that matches visemes to words and predicts the uttered sentence using perplexity analysis. The performance of the system is evaluated by comparing the sentences predicted by the lip reading system to the ground truth of the spoken sentences and measuring the edit distance. In the following Sections, details of the systems components are discussed.

A. ARCHITECTURE

The overall system used for decoding speech consists of two separate neural network architectures used to perform two different tasks. The first architecture is used for the task of viseme classification and consists of a spatial-temporal visual frontend in tandem with an attention-based transformer and the predicted visemes provide the input of the next architecture. The second architecture, also an attention-based transformer, is used to predict the spoken words given the uttered visemes using a calculated metric called perplexity. As illustrated in Figure 2, each of these modules are briefly described along with the overall framework for the lip reading system. Both the viseme classifier and the word detector consist of common blocks including fully connected layers, self-attention layers and feed-forward layers and the breakdown of these three blocks is given in Figure 3.

The attention-transformer structure used in [40] has been changed to fit visemes, and this will be discussed in III-E. Unlike [40], there is no embedding layer, and the Decoder has been altered with the final softmax layer trained on visemes instead of ASCII characters.

B. DATA

The dataset used in this research is the BBC LRS2 dataset [6]. It consists of approximately 46,000 videos covering over 2 million word instances and a vocabulary range of over 40,000 words. The video with the longest duration has a length of 180 frames with every video have frame rate of 25 frames per second. The dataset contains sentences of up to 100 ASCII characters from BBC videos, with a range of facial poses from frontal to profile. The dataset is extremely difficult due to the variety of viewpoints, lighting conditions, genres and the number of speakers.

Table 2 gives a breakdown of the different sections of the BBC LRS2 data with statistics of how many sentences there are, the number of word instances, the vocabulary range and the ratio of profile to frontal videos in that particular section of the corpus.

TABLE 2: Statistics of BBC LRS2 dataset.

Split	Utterances	Word Instances	Vocabulary	Frontal/Profile Split (%)
Train	45839	329180	17660	64.8:35.2
Test	1243	6660	1697	63.5:36.5

C. DATA PRE-PROCESSING

All the videos are pre-processed according to the stages given in Figure 4. Videos consist of images with red, green and blue pixel values and resolution 160 pixels by 160 pixels; with a frame rate of 25 frames/second. Videos are first sampled into image frames, then once the videos are sampled, facial landmarks need to be located as the speaking person's lips are the region of interest and feature input to the visual frontend. The Single Shot MultiBox Detector (SSD) [46], a CNN-based detector, is used for detecting face appearances within the individual frames and to recognise facial landmarks according to the iBug [47] landmark convention of 68 landmarks, and it

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3040906, IEEE Access

FIGURE 1: The components of the overall lip reading system.

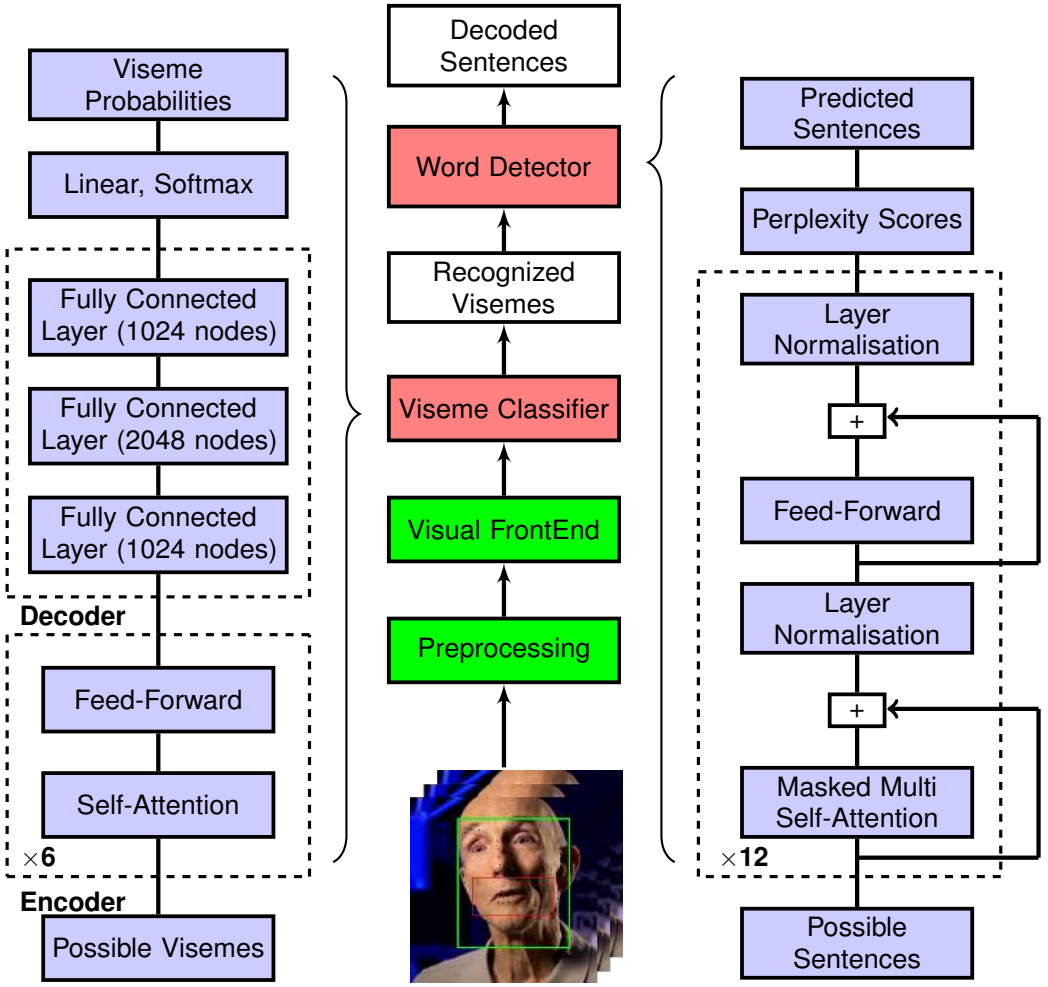

FIGURE 2: The breakdown stages of how sentences are predicted from silent videos.

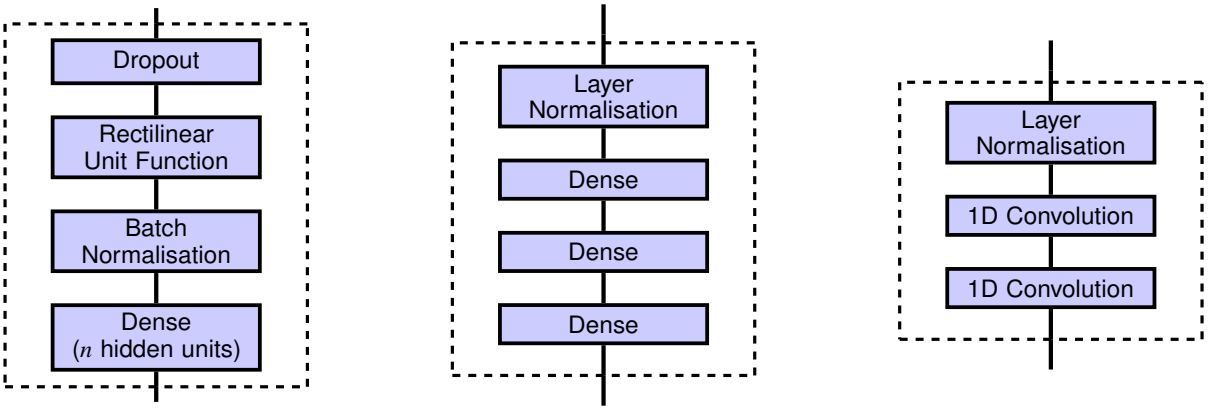

FIGURE 3: The different transformer components with the fully connected layer on the **left**, self-attention in the **middle** and feed-forward on the **right**.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3040906, IEEE Access

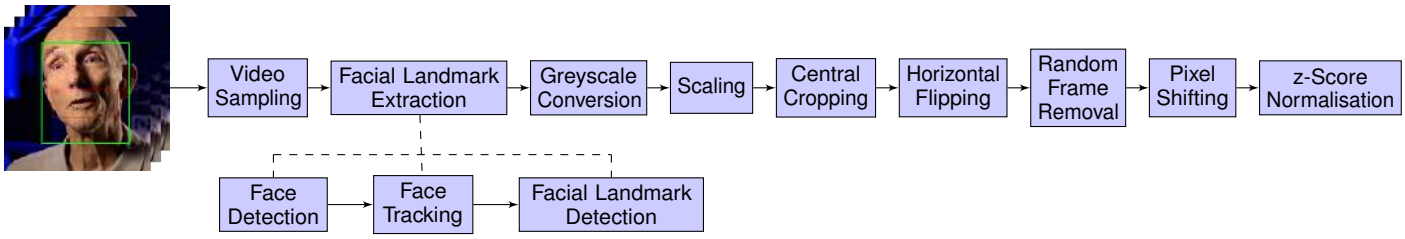

FIGURE 4: The stages of video image pre-processing.

can be used on faces pointing at different angles. Landmarks are applied according to the stages shown in Figure 5 with the face detected shown on the left, the face being tracked in the middle and where facial landmarks are detected on the right.

The video frames are then converted to greyscale, scaled, and then centrally cropped around the boundary of the facial landmarks resulting in reduced image dimensions of $112 \times 112 \times T$ dimensions (where T corresponds to the number of image frames). Data augmentation in the form of horizontal flipping, removal of random frames [38], [39], and random shifts of up to ± 5 pixels in the spatial dimension and ± 2 frames in the temporal dimension respectively, respectively, are also applied. At the end, pixels are normalized with respect to the overall mean and variance of every pixel in each frame.

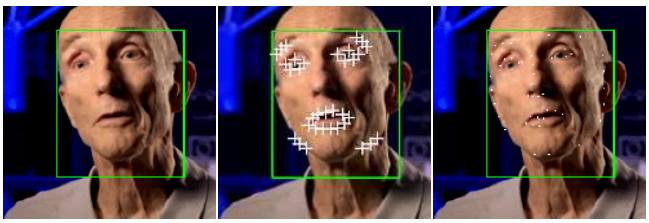

FIGURE 5: The three substages of Facial Landmark Extraction with face detection on the **left**, face tracking in the **middle** and facial landmark detection on the **right**.

Pre-processing is needed in order to ensure that the appropriate region of interest (ROI) can be extracted as the input to the neural network with resolution 112×112 pixels that contains the lips. The ROI must also undergo greyscale conversion and z-score normalization. The facial landmark detection described earlier has already been performed on every single video contained within the BBC LRS2 corpus. Some of the pre-processing steps described in Figure 4 may not be necessary for this corpus, as the 112×112 set of pixels can be extracted through central cropping of the original image frames with 160×160 pixels. The entire pre-processing process would however be a necessity for a lip reading system that can be generalized to other real-time applications.

D. VISUAL FRONTEND

The spatial-temporal visual front-end is based on [39]. The network applies a spatial-temporal (3D) convolution on the input image sequence, with a filter width of five frames, followed by a 2D ResNet that gradually decreases the spatial dimensions with depth. For an input sequence of $T \times H \times W$ frames, the output is a $T \times \frac{H}{32} \times \frac{W}{32} \times 512$ tensor (i.e., the temporal resolution is preserved) and it is then average-pooled over the spatial dimensions, yielding a 512-dimensional feature vector for every input video frame. Details of the architecture for the Visual Frontend are given in Table 3. The trained network used in [8] has been applied in this work.

TABLE 3: Details of spatial-temporal network for visual front-end.

Layer Type	Filter	Output Dimensions
3D Convolution	$[5 \times 7 \times 7, 64]$/(1,2,2)	$180 \times 56 \times 56 \times 64$
3D Max Pooling	(1,2,2)	$180 \times 28 \times 28 \times 64$
Residual 2D Convolution	$[3 \times 3, 64] \times 2/(1, 1)$	$180 \times 28 \times 28 \times 64$
Residual 2D Convolution	$[3 \times 3, 64] \times 2/(1, 1)$	$180 \times 28 \times 28 \times 64$
Residual 2D Convolution	$[3 \times 3, 128] \times 2/(2, 2)$	$180 \times 14 \times 14 \times 128$
Residual 2D Convolution	$[3 \times 3, 128] \times 2/(1, 1)$	$180 \times 14 \times 14 \times 128$
Residual 2D Convolution	$[3 \times 3, 256] \times 2/(2, 2)$	$180 \times 7 \times 7 \times 256$
Residual 2D Convolution	$[3 \times 3, 256] \times 2/(1, 1)$	$180 \times 7 \times 7 \times 256$
Residual 2D Convolution	$[3 \times 3, 512] \times 2/(2, 2)$	$180 \times 4 \times 4 \times 512$
Residual 2D Convolution	$[3 \times 3, 512] \times 2/(1, 1)$	$180 \times 4 \times 4 \times 512$

E. VISEME CLASSIFIER

Lip reading datasets consist of labels in the form of subtitles. These subtitles are strings of words that need to be converted to sequences of visemes to provide labels for the viseme classifier. The conversion is performed in two stages: first, they are mapped to phonemes using the Carnegie Mellon Pronouncing Dictionary [41], and then the phonemes are mapped to visemes according to Lee and Yook's approach [30]. Table 4 shows the mapping. The attention transformer which predicts the spoken visemes from a person speaking in a silent video uses 17 classes in total; these include the 13 visemes, a space character, start of sentence (SoS), end of sentence (EoS) and a character for padding. All the defined classes are listed in Table 5. All videos are padded to 180 characters.

The Transformer [40] model has an encoder-decoder structure with multi-head attention layers used as building blocks. The encoder used is a stack of self-attention layers, where the input tensor serves as the attention queries, keys and values at the same time. The decoder here consists of 3 fully connected layer blocks structured as shown in Figure 6; and each fully connected layer blocks consists of a dense layer, batch normalisation, rectilinear unit function and a dropout layer of probability 0.1. The dense layer within the middle fully connected layers consists of 2048 nodes while the dense layers within the first and last fully connected layer blocks

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3040906, IEEE Access

only contain 1024 nodes. The decoder produces character probabilities which are directly matched to the ground truth labels and trained with a cross-entropy loss. The encoder follows the base model of [40] with 6 layers, model size 512, 8 attention heads and dropout with probability 0.1.

TABLE 4: Viseme-to-Phoneme Mappings.

Viseme Class	Viseme Type	Phonemes Set
p	consonant	b, p, m
t	consonant	d, t, s, z, th, dh
k	consonant	g, k, n, ng, l, y, hh
ch	consonant	jh, ch, sh, zh
f	consonant	f, v
w	consonant	r, w
iy	vowel	iy, ih
ey	vowel	eh, ey, ae
aa	vowel	aa, aw, ay, ah
ah	vowel	ah
ao	vowel	ao, oy, ow
uh	vowel	uh, uw
er	vowel	er
s	silent character	sil

TABLE 5: Classes used by Viseme Classifier.

[pad], AA, AH, AO, CH, ER, EY, F, IY, K, P, T, UH, W, <sos>, <eos>, [space]

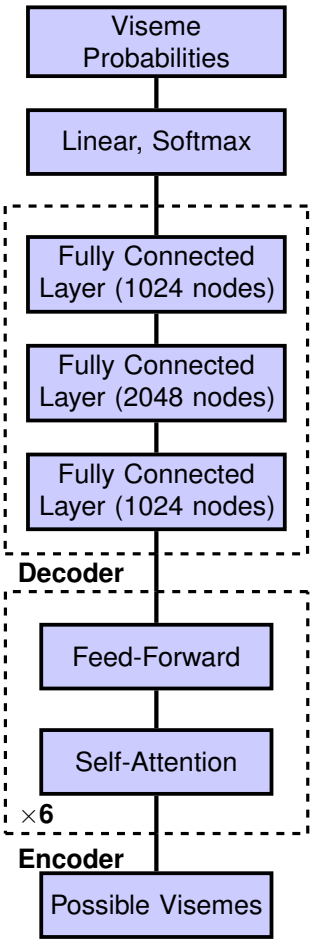

FIGURE 6: The architecture of transformer for the Viseme Classifier.

However, it should be noted that the decoder utilised in this work follows a completely different structure from that of [8] for the following reasons:

1) There are no embeddings;
2) The predicted labels from the previous timestep are not fed into the decoder as it is assumed that visemes do not have the conditional probability relationship that ASCII characters have. This means that no teacher forcing is used whereby the ground truth of the previous decoding step has to be supplied as the input to the decoder.; and
3) It is only the decoder and dense layer that differ, so the trained weights from [8] have been used and applied to both the visual frontend and encoder, where only the decoder layers and dense layers are trained.

Because the encoder has an identical topology to that used by [8], the trained weights from their model have been applied to here and it is only the decoder and the final softmax layer in Figure 6 that are to be trained. During the training phase, the Adam optimiser [44] is used with default parameters and initial learning rate 10^{-3}, reducing it on plateau down to 10^{-4} and all operations are implemented in TensorFlow and trained on a single GeForce GTX 1080 Ti GPU with 11GB memory.

F. WORD DETECTOR

The outputted visemes from the viseme classifier need to be further converted to meaningful sentences or strings of words. Every word in a sentence contains a set of visemes and therefore can be mapped to a cluster of visemes, such that a cluster of visemes is a set of visemes which make up a word. Once visemes have been classified, the viseme-to-word conversion process needs to be performed. Because a cluster of visemes can map to several different words, the combination of the words that were uttered by the speaker still needs to be deciphered. The solution to the problem is to select the most likely combination of words. The general procedure for converting visemes to words with different stages is given in Figure 7.

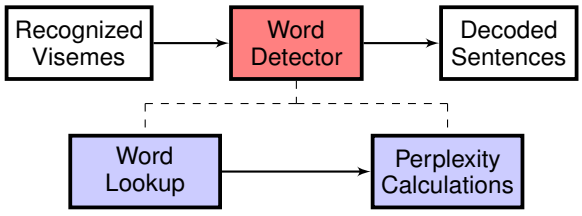

FIGURE 7: The components of the Word Detector.

The first stage of the Word Detection is the World Lookup stage. Every single cluster of visemes needs to be mapped to a set of words containing those visemes according to the mapping given by the Carnegie Mellon Pronouncing (CMU) Dictionary. However, if there are clusters where no match is found, a cluster in the dictionary that most closely resembles it is used instead and the words mapping to that cluster are used. The resemblance is determined using Levenshtein distance [22] and the cluster in the CMU dictionary with the smallest value is chosen.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3040906, IEEE Access

Once the word lookup stage is performed, the next stage of Word Detection is the Perplexity Calculations. The different possible choices of words that map to the visemes are combined, and perplexity iterations are performed to determine which combination of words is most likely to correspond to the uttered sentence, given the visemes recognised. Naturally, the sentence that is most grammatically correct will have the highest likelihood [45] and perplexity is one metric that can be used to compare sentences to determine which is most grammatically sound. The rationale behind perplexity is discussed later with an even more detailed description about how perplexity analysis is used to convert viseme to words. In this paper the following rules are used when predicting sentences and they are based on determining which combinations of words have the greatest likelihood according to probabilistic information theory:

1) If a viseme sequence has only 1 cluster matching to one word, that one word is selected as the output.
2) If a viseme sequence has only 1 cluster matching to several words, that word with largest expectation is selected as the output.
3) If a viseme sequence has more than 1 cluster, the words matching to the first two clusters are combined in every possible combination for the first iteration.
    a) The combinations with the lowest 50 perplexity scores are kept.
    b) These combinations are in turn combined with the words matching to the next viseme cluster.
    c) The combinations with the lowest 50 perplexity scores are kept and the iterations continue for the remaining clusters of the sequence until the end of the sequence is reached.

The selection of the lowest 50 perplexity scores at each iteration is based on an implementation of a local beam search with width 50. In practice, it would be computationally expensive to do an exhaustive search so a beam search has been implemented to reduce the computational overhead, and the beam width is an arbitrary figure chosen as a compromise between accuracy and computational efficiency.

Eqs. 1 to 4 below describe the probabilistic relationship between the observed visemes and the words spoken; where V is the spoken sequence of viseme clusters, v_i corresponds to every ith cluster, W_C represents any given combination of words and w_i corresponds to every ith word within the string of words. The string of words $\check{W}$ that is to be selected will be the combination that has the maximum likelihood given the identity of the viseme clusters for every combination C that falls within the set of combinations C^*. The sequence of visemes clusters given in Eq. 1 maps to any possible combination of words as given in Eq. 2, and the solution to predicting the sentence spoken is the combination of words given the recognised visemes which has the greatest probability as expressed in Eqs. 3 and 4.

$$V = (v_1, v_2, ..., v_N) = \sum_{i=1}^{N} v_i \tag{1}$$

$$W_C = (w_1, w_2, ..., w_N)_C = \sum_{i=1}^{N} w_i \tag{2}$$

$$\check{W} == \arg\max_{C \epsilon C^*} [P(W|V)]_C \tag{3}$$

$$\check{w}_1, \check{w}_2, ..., \check{w}_N = \arg\max_{C \epsilon C^*} [P(w_1, w_2, ..., w_N | v_1, v_2, ..., v_N)]_C \tag{4}$$

If the identity of observed visemes is known, the probability of the viseme sequence in Eq. 1 is equal to 1, resulting in the expression in Eq. 5. The choice of words predicted according to Eq. 4 gets reduced to the expression given in Eq. 6.

$$P(v_1, v_2, , v_N) = 1 \tag{5}$$

$$\check{w}_1, \check{w}_2, ..., \check{w}_N = \arg\max_{C \epsilon C^*} [P(w_1, w_2, ..., w_N)]_C \tag{6}$$

Eqs. 7 to 10 below describe the relationship between the perplexity PP, entropy H and probability $P(w_1, w_2, ..., w_N)$ of a particular sequence of N words $(w_1, w_2, ..., w_N)$. The word detector consists of a trained attention-based transformer for calculating PP expressed as the exponentiation of H in Eq. 7. The per-word entropy $\hat{H}$ is related to the probability $P(w_1, w_2, ..., w_N)$ of words $(w_1, w_2, ..., w_N)$ belonging to a vocabulary set W, and is calculated as a summation over all possible sequences of words. If the source is ergodic, the expression for $\hat{H}$ in Eq. 8 gets reduced to that in Eq. 9. The value of $P(w_1, w_2, ..., w_N)$ resulting in the choice of words selected as the output for Eq. 6 also results in the minimisation of entropy in Eq. 9, further resulting in the minimisation of perplexity given in Eq. 10.

$$PP = e^H \tag{7}$$

$$\hat{H} = -\lim_{N \to \infty} \frac{1}{N} \sum_{w_1, w_2, ... w_N} P(w_1, w_2, ..., w_N) \ln P(w_1, w_2, ..., w_N) \tag{8}$$

$$\hat{H} = -\frac{1}{N} \ln P(w_1, w_2, ..., w_N) \tag{9}$$

$$PP = P(w_1, w_2, ..., w_N)^{-\frac{1}{N}} \tag{10}$$

A language model, i.e., a probability distribution over sequences of words, can be measured on the basis of the entropy of its output from the field of information theory [43]. Perplexity is a measure of the quality of a language model, because a good language model will generate sequences of words with a larger probability of occurrence resulting in a smaller perplexity.

The Transformer model used for the word detector is the pre-trained Generative Pre-Training (GPT) Transformer [42] - a multi-layer decoder and a variant of the transformer used in [40]. It consists of repeated blocks of multi-headed self-attention followed by position-wise feedforward layers. The architecture is typically used for sentence prediction; however,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3040906, IEEE Access

the architecture itself here is not used for direct classification, rather its purpose is for perplexity calculations that are required for word selection where visemes are converted to words. Visemes from the previous step are sequentially matched to words and the most probable sentence is chosen according to that with the minimum perplexity score. The perplexity score is calculated by taking the exponentiation of the cross-entropy loss when the GPT is evaluated on a sentence and like in [27], a beam width of 50 has been used.

G. SYSTEMS PERFORMANCE MEASURES

The measures that have been used to evaluate the lip reading sentence system are edit distance-based metrics and are computed by calculating the normalized edit distance between the ground truth and a predicted sentence. Metrics reported in this paper include Viseme Error Rate (VER), Character Error Rate (CER), Word Error Rates (WER) and Sentence Accuracy Rate (SAR).

Error rate metrics used for evaluating accuracy are given by calculating the overall edit distance. In determining misclassifications, one has to compare the decoded speech to the actual speech. The equation for calculating Error Rate (ER) is given in Eq. 11 with N being the total number of characters in the ground truth, S being the number of characters substituted for wrong classifications, I being the number of characters inserted for those not picked up and D being the number of deletions being made for decoded characters that should not be present. CER, WER and VER are all calculated this way with the expressions given in Eqs. 12, 13 and 14 where C, W and V correspond to characters, words and visemes.

$$ER = \frac{S + D + I}{N} \tag{11}$$

$$CER = \frac{C_S + C_D + C_I}{C_N} \tag{12}$$

$$WER = \frac{W_S + W_D + W_I}{W_N} \tag{13}$$

$$VER = \frac{V_S + V_D + V_I}{V_N} \tag{14}$$

SAR is a binary metric as expressed in Eq. 15, where the value is 1 if the predicted sentence P_P is equal to the ground truth P_T, otherwise it would take the value of 0:

$$SAR = \begin{cases} 1, & P_P = P_T \\ 0, & P_P \neq P_T \end{cases} \tag{15}$$

H. ILLUMINATION

To test the proposed lip reading system's robustness to changes in lighting, the overall architecture, once trained, has been evaluated on videos from the testing set under levels of illumination. Illumination has been applied by varying the pixel brightness. It is after the video sampling stage of the pre-processing described in III-C that illumination is applied to the image frames. The overall process is described in Figure 8.

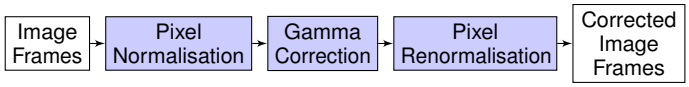

FIGURE 8: Stages for applying illumination.

Image frames of videos from the dataset consist of red, blue and green pixel components with numerical values ranging from minimum intensity 0 to maximum intensity 255. Pixel normalisation is the first stage of the procedure and this involves minimum-maximum normalisation of all pixel values where pixel values are mapped from the range [0,255] to [0,1]. Once this is done, a gamma correction is applied where pixel values are corrected according to Eq. 16, where I is a matrix of pixels, γ is scalar value and O is the resulting matrix of pixels after the gamma correction has been applied:

$$O = I^{1/\gamma} \tag{16}$$

Values of γ that are less than 1.0 will cause images to darken whereas values of γ that are greater than 1.0 cause images to brighten. Figure 9 gives examples of images with the standard image ($\gamma = 1.0$) on the left, the darkened image in the middle ($\gamma = 0.5$) and the brightened image on the right ($\gamma = 1.5$). The gamma corrections applied in this paper have utilised γ values ranging from 0.5 to 1.5.

FIGURE 9: Images under varying illumination with standard image on the **left**, darkened image in the **middle** and brightened image on the **right**.

After applying the gamma correction, pixels undergo renormalisation where all pixels values are mapped back from from the range [0,1] to the range [0,255].

IV. EXPERIMENTS AND RESULTS

For training and evaluation of the viseme classifier, the BBC LRS2 dataset described in III-B has been used with 45839 sentences for training and 1243 sentences for testing. All components of the model are evaluated on the LRS2 test set. The metrics reported include VER, CER, WER, SAR and the total overall training time.

The viseme classifier was trained for a total of 2000 epochs and it was at the point that the validation loss started to become saturated, and when no further convergence was recorded that the model was evaluated. Plots for the loss and VER for both training and validation are given in Figures 10 and 11.

The results are summarized in Table 6. As shown in the Table, the overall WER of 35.4% is a reduction of almost 15% compared to the 50% achieved in a previous state-of-the-art

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3040906, IEEE Access

model trained and evaluated on the same dataset; and thus, improvement on the overall word accuracy to 64.6%. The accuracy by visemes was also very high with a VER of only 4.6%. The confusion matrices by both visemes and ASCII characters are given in Figures 12 and 13, respectively.

Table 7 gives the performance metrics for how the proposed lip reading system and Afouras et al's model [8] performed when videos in the validation set were subjected to different levels of illumination, applied to in accordance with III-H. It can be seen that the proposed lip reading system is generally robust to varying levels of illumination, like that of Afouras et al [8] and this is expected given that videos in the BBC LRS2 corpus were recorded in varying lighting conditions.

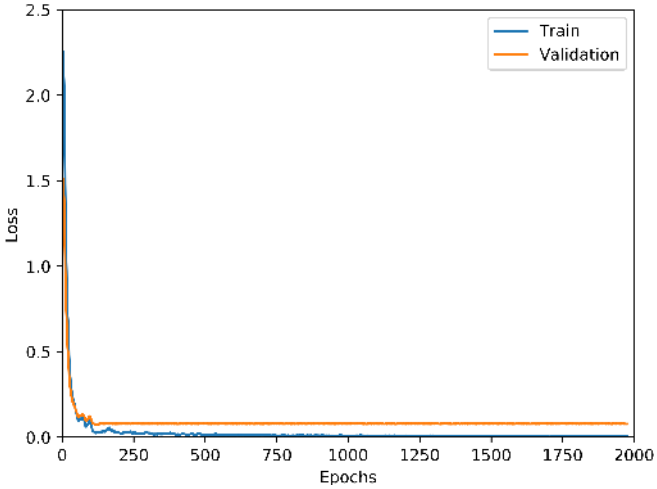

FIGURE 10: Loss curve for training and validation.

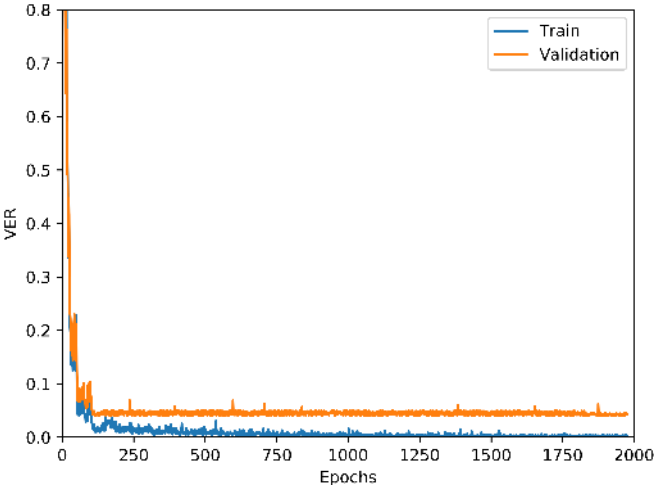

FIGURE 11: VER curve for training and validation.

TABLE 6: The performance results of lip reading sentences.

Validation Samples	Parameters	VER(%)	CER(%)	WER(%)	SAR(%)	CPU Time
1243	4,748,305	4.6	23.1	35.4	33.4	37 hours

TABLE 7: The performance of proposed system under varying illumination.

Gamma	Visual Lip Reading System			Afouras et al.		
	VER(%)	WER(%)	SAR(%)	CER(%)	WER(%)	SAR(%)
0.5	5.4	41.5	21.8	35.8	53.9	18.4
0.8	5.0	37.9	28.5	33.9	51.0	20.3
0.9	4.7	35.7	32.7	33.7	50.9	20.6
1	4.6	35.4	33.4	33.7	50.8	20.8
1.1	4.7	35.6	32.9	33.7	50.8	20.2
1.2	4.9	37.4	29.4	34.1	51.4	20.6
1.5	5.3	40.5	23.7	36.2	51.4	20.2

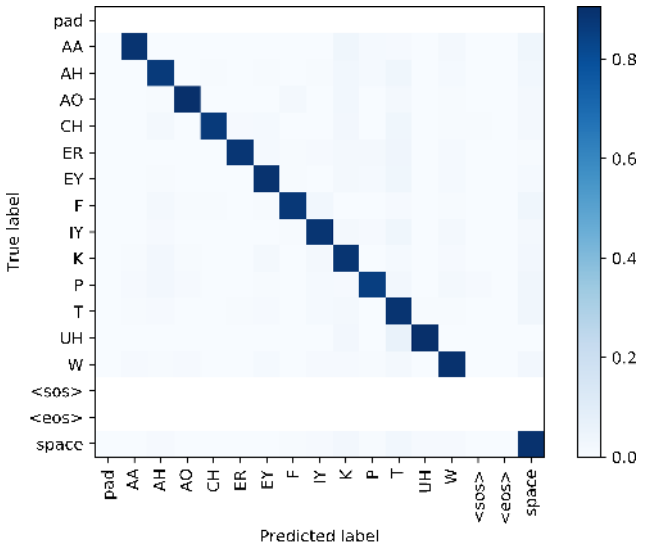

FIGURE 12: Confusion matrix for classification of visemes.

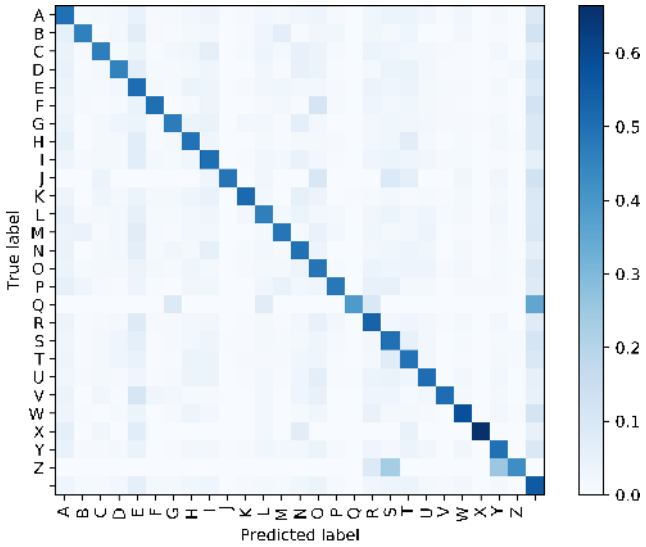

FIGURE 13: Confusion matrix for classification of ASCII characters.

In order to attain a good overall accuracy for classification of words, both the viseme classification performance and the viseme-to-word conversion performance need to be good. The VER is very low and any misclassifications that have occurred during the validation phase appeared to be influenced by the class imbalance of visemes present in the training data. When visemes are misclassified, they are most likely to be decoded as one of "AH", "K" or "T" because such visemes appear most frequently in training data and obscure classes such as "AA" and "CH" are the most likely to be misclassified.

Table 8 gives examples of sentences from the BBC LRS2 dataset along with the decoded visemes, the word combinations that were outputted at each iteration of the perplexity calculations, and the viseme clusters corresponding to each predicted word. Table 9 gives the full details of how those sentences were decoded by listing their corresponding visemes, the predicted visemes, the decoded sentences and their corresponding metric performance results.

A stratified sampling strategy was used to select the most frequently appearing 154 words in the BBC LRS2 training set that begin with each letter of the alphabet. For the selected 154 words, a comparison of the accuracy in terms of ratio of how many times a word was correctly decoded to how many times it appeared in the testing phase has been presented in Figures 14 and 15. Figure 14 shows the word accuracy for Afouras et al.'s model and Figure 15 shows the accuracy for this lip reading system. A better word precision is noticeable in Figure 15.

It should be noted that, whilst the VER was low, the WER was still high although it has been significantly improved compared to other existing works. To further reduce the error rate, the viseme-to-word conversion would need to be optimised. Many misclassifications have been caused by the presence of local optima during the implementation of the local beam search, whereby at each iteration of the viseme sequence during the perplexity calculation stage, the words that make up the ground truth are not included within the top 50 results. A large beam with would invariably result in a greater conversion rate, but at the expense of using more computational overhead and an exhaustive search would not even be viable. Further work needs to be done to ensure that the global optimum combinatorial solution is selected more frequently during the Perplexity Calculation stage to further improve on word accuracy.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3040906, IEEE Access

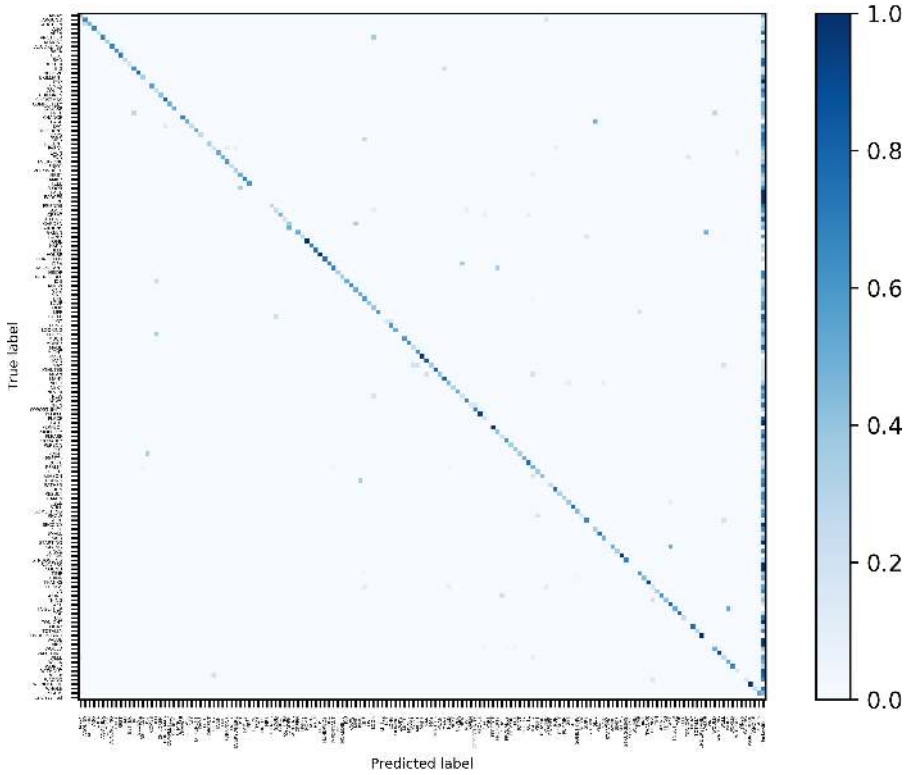

FIGURE 14: Word confusion matrix for Afouras et al's model.

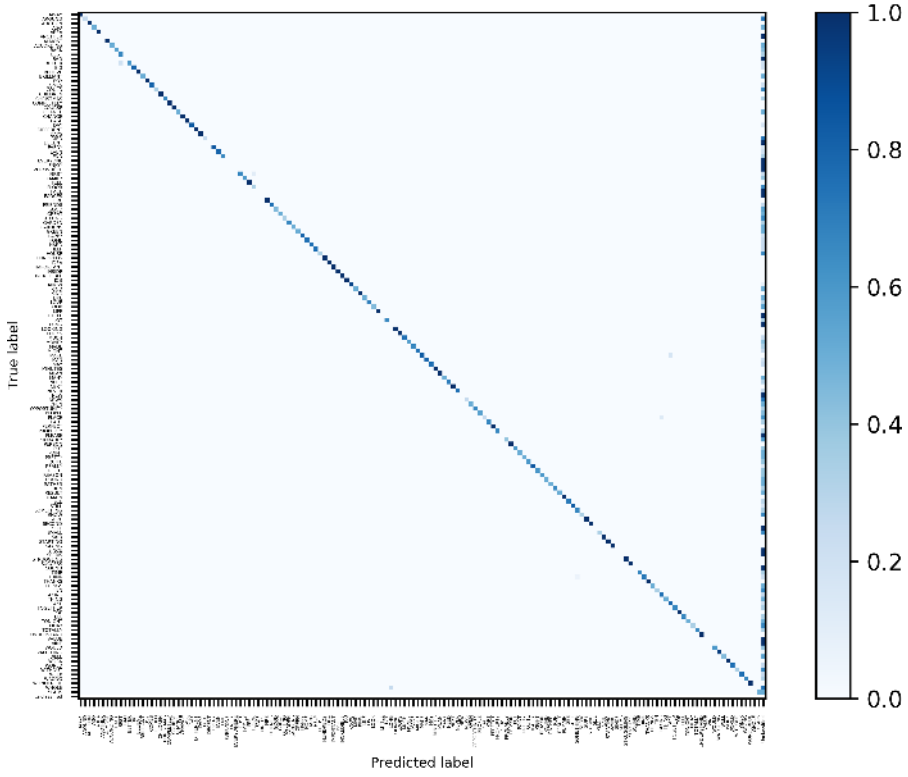

FIGURE 15: Word confusion matrix for this lip reading system.

TABLE 8: Examples of perplexity calculations for sentences from the test set.

Actual Subtitle	Predicted Visemes	Decoded Subtitle Perplexity	Word Correspondence
I	(’AH’),	(’a nouns’, 161.9), ("i can’t", 184.3), (’uh nouns’, 204.3), ...	(’AH’): i
CAN’T	(’K’, ’EY’,’K’,’T’),	(’a nouns but’, 182.5), (’uh nouns but’, 200.9), ("i can’t but", 223.3), ...	(’K’, ’EY’,’K’,’T’): can’t
PUT	(’P’,’AH’,’T’),	(’a nouns but it’, 120.6), (’uh nouns but it’, 125.0), ...	(’P’,’AH’,’T’): bite
IT	(’IY’,’T’),	("i can’t bite it any", 181.8), ("i can’t buss it any", 242.2), ...	(’IY’,’T’): it
ANY	(’EY’,’K’,’IY’),	("i can’t bite it any plainer", 130.8), ("i can’t buss it any plainer", 183.4), ...	(’EY’,’K’,’IY’): any
PLAINER	(’P’,’K’,’EY’,’K’,’ER’),	("i can’t bite it any plainer than", 87.4), ...	(’P’,’K’,’EY’,’K’,’ER’): plainer
THAN	(’T’,’EY’,’K’),	("i can’t bite it any plainer than that", 57.7), ...	(’T’,’EY’,’K’): than
THAT	(’T’,’EY’,’T’)	Result: i can’t bite it any plainer than that	(’T’,’EY’,’T’): that
WHEN	[’W’,’EY’,’K’),	(’when there’, 121.0), ("when they’re", 216.4), (’whack their’, 220.6), ...	[’W’,’EY’,’K’): when
THERE	(’T’,’EY’,’W’),	("when they’re isn’t", 69.9), ("wreck there isn’t", 88.9), ...	(’T’,’EY’,’W’): there
ISN’T	(’IY’,’T’,’AH’,’K’,’T’),	("when there isn’t much else", 52.5), ...	(’IY’,’T’,’AH’,’K’,’T’): isn’t
MUCH	(’P’,’AH’,’CH’),	("when there isn’t much else in", 60.9), ...	(’P’,’AH’,’CH’): much
ELSE	(’EY’,’K’,’T’),	("when there isn’t much else in the", 41.9), ...	(’EY’,’K’,’T’): else
IN	(’IY’,’K’),	("when there isn’t much else in the garden", 60.3), ...	(’IY’,’K’): in
THE	(’T’,’AH’),	Result: when there isn’t much else in the garden	(’T’,’AH’): the
GARDEN	(’K’,’AA’,’W’,’T’,’AH’,’K’)		(’K’,’AA’,’W’,’T’,’AH’,’K’):garden
SORT	(’T’,’AO’,’W’,’T’),	(’sort of’, 1.2), (’source of’, 1.5), (’doors of’, 25.0), (’sword of’, 28.3), ...	(’T’,’AO’,’W’,’T’): sort
OF	(’AH’,’F’),	(’sort of second’, 55.3), (’sort of talent’, 81.1), (’source of talent’, 89.3), ...	(’AH’,’F’): of
SECOND	(’T’,’EY’,’K’,’AH’,’K’,’T’),	(’sort of tennent naff’, 147.4), (’sort of second half’, 158.6), ...	(’T’,’EY’,’K’,’AH’,’K’,’T’): second
HALF	(’K’,’EY’,’F’),	(’sort of second half of’, 60.4), ("zorz i’ve tennent naff i’ve", 132.7), ...	(’K’,’EY’,’F’): half
OF	(’AH’,’F’),	(’sort of second half of october’, 229.1), ...	(’AH’,’F’): of
OCTOBER	(’AA’,’K’,’T’,’AO’,’P’,’ER’)	Result: sort of second half of october	(’AA’,’K’,’T’,’AO’,’P’,’ER’): october
BUT	(’W’,’AH’,’T’),	(’right before’, 188.2), (’ride before’, 309.3), (’rise before’, 319.8), ...	(’W’,’AH’,’T’): right
BEFORE	(’P’,’IY’,’F’,’AO’,’W’),	(’right before i’, 41.2), (’ride before i’, 69.1), (’ries before i’, 81.2), ...	(’P’,’IY’,’F’,’AO’,’W’): before
I	(’AH’),	(’right before i do’, 55.9), (’ride before i do’, 78.6), ...	(’AH’): i
DO	(’T’,’UH’)	Result: right before i do	(’T’,’UH’): do
AS	(’IY’,’T’),	(’is a’, 14.4), (’eat a’, 39.7), (’ease a’, 56.6), ("e.’s a", 132.8), ...	(’IY’,’T’): is
A	(’AH’),	("e’s a whittle’s", 157.6), ("e’s i. whittle’s", 191.8),...	(’AH’): a
RESULT	(’W’,’IY’,’T’,’AH’,’K’,’T’),	(’is a result of’, 40.0), ("e’s i. whittle’s i’ve", 106.4), ...	(’W’,’IY’,’T’,’AH’,’K’,’T’): result
OF	(’AH’,’F’),	(’is a result of smoking’, 135.4), ("e’s a whittle’s i’ve smolin", 190.9), ...	(’AH’,’F’): of
SMOKING	(’T’,’P’,’AO’,’K’,’IY’,’K’)	Result: is a result of smoking	(’T’,’P’,’AO’,’K’,’IY’,’K’): smoking
PRETTY	(’W’,’W’,’IY’,’T’,’IY’),	(’wheatie on’, 169.2), (’reidy on’, 296.3), (’riedy on’, 349.9), ...	(’W’,’W’,’IY’,’T’,’IY’): witty
ON	(’AA’,’K’),	(’weedy on the’, 29.4), (’reedy on the’, 31.8), (’witty on the’, 56.7), ...	(’AA’,’K’): on
THE	(’T’,’AH’),	(’witty on the outside’, 45.3), (’weedy on the outside’, 52.2), ...	(’T’,’AH’): the
OUTSIDE	(’EY’,’T’,’T’,’AH’,’T’)	Result: witty on the outside	(’EY’,’T’,’T’,’AH’,’T’): outside
EVEN	(’T’,’AH’,’IY’,’K’),	(’dying before’, 346.6), (’sighing before’, 368.7), ...	(’T’,’AH’,’IY’,’K’): dying
BEFORE	(’P’,’IY’,’F’,’AO’,’W’),	(’sighing before she’, 64.1), (’dying before she’, 77.1), ...	(’P’,’IY’,’F’,’AO’,’W’): before
SHE	(’CH’,’IY’),	(’sighing before she answered’, 35.3), ...	(’CH’,’IY’): she
ENTERED	(’EY’,’K’,’T’,’ER’,’T’),	(’sighing before she answered a’, 85.4), ...	(’EY’,’K’,’T’,’ER’,’T’): entered
THE	(’AH’),	(’dying before she entered a water’, 190.5), ...	(’AH’): a
WATER	(’W’,’AO’,’T’,’ER’)	Result: dying before she entered a water	(’W’,’AO’,’T’,’ER’): water
LIKE	(’K’,’AH’,’K’),	(’nine hundreds’, 1831.3), (’lysne hundreds’, 2486.6), ...	(’K’,’AH’,’K’): nine
HUNDREDS	(’K’,’AH’,’K’,’T’,’W’,’AH’,’T’,’T’),	(’nine hundreds of’, 62.7), (’cul hundreds of’, 113.9), ...	(’K’,’AH’,’K’,’T’,’W’,’AH’,’T’,’T’): hundreds
OF	(’AH’,’F’),	(’nine hundreds of thousands’, 49.2), ...	(’AH’,’F’): of
THOUSANDS	(’T’,’EY’,’T’,’AH’,’K’,’T’,’T’),	(’nine hundreds of thousands of’, 20.8), ...	(’T’,’EY’,’T’,’AH’,’K’,’T’,’T’): thousands
OF	(’AH’,’F’),	(’nine hundreds of thousands of peopled’, 72.3), ...	(’AH’,’F’): of
PEOPLE DO	(’P’,’IY’,’P’,’AH’,’K’,’K’,’T’,’UH’),	(’nine hundreds of thousands of peopled every’, 103.8), ...	(’P’,’IY’,’P’,’AH’,’K’,’K’,’T’,’UH’): peoples
EVERY	(’EY’,’F’,’ER’,’IY’),	(’nine hundreds of thousands of peoples every year’, 65.4), ...	(’EY’,’F’,’ER’,’IY’): every
YEAR	(’K’,’IY’,’W’)	Result: nine hundreds of thousands of peoples every year	(’K’,’IY’,’W’) year

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3040906, IEEE Access

TABLE 9: Examples of how sentences from the test set were decoded.

Actual Subtitle	Corresponding Visemes	Predicted Visemes	Decoded Subtitle	VER(%)	CER(%)	WER(%)	SAR(%)
I CAN'T PUT IT ANY PLAINER THAN THAT	[('AH'),('K','EY','K','T'), ('P','UH', 'T'),('IY','T'), ('EY','K','IY'), ('P','K','EY', 'K','ER'), ('T','EY','K'),('T','EY','T')]	('AH'),('K', 'EY','K','T'), ('P','AH','T'),('IY','T'), ('EY','K','IY'), ('P','K','EY','K','ER'), ('T','EY','K'),('T','EY','T')	I CAN'T BITE IT ANY PLAINER THAN THAT	3.1	8.3	12.5	0.0
WHEN THERE ISN'T MUCH ELSE IN THE GARDEN	('W','EY','K'),('T','EY','W'), ('IY','T','AH','K','T'), ('P','AH','CH'),('EY','K','T'), ('IY','K'),('T','AH'), ('K','AA','W','T','AH','K')	['W','EY','K'),('T','EY','W'), ('IY','T','AH','K','T'), ('P','AH','CH'),('EY','K','T'), ('IY','K'),('T','AH'), ('K','AA','W','T','AH','K')	WHEN THERE ISN'T MUCH ELSE IN THE GARDEN	0.0	0.0	0.0	100.0
SORT OF SECOND HALF OF OCTOBER	('T','AO','W','T'),('AH','F'), ('T','EY','K','AH','K','T'), ('K','EY','F'), ('AH','F'), ('AA','K','T','AO','P','ER')	('T','AO','W','T'),('AH','F'), ('T','EY','K','AH','K','T'), ('K','EY','F'), ('AH','F'), ('AA','K','T','AO','P','ER')	SORT OF SECOND HALF OF OCTOBER	0.0	0.0	0.0	100.0
BUT BEFORE I DO	[('P','AH','T'), ('P','IY','F','AO','W'), ('AH'),('T','UH')]	('W','AH','T'), ('P','IY','F','AO','W'), ('AH'),('T','UH')	RIGHT BEFORE I DO	6.7	26.7	25.0	0.0
AS A RESULT OF SMOKING	('EY','T'),('AH'), ('W','IY','T','AH','K','T'), ('AH','F'), ('T','P','AO','K','IY','K')	('IY','T'),('AH'), ('W','IY','T','AH','K','T'), ('AH','F'), ('T','P','AO','K','IY','K')	IS A RESULT OF SMOKING	4.5	4.5	20.0	0.0
PRETTY ON THE OUTSIDE	('P','W','IY','T','IY'),('AA','K'), ('T','AH'),('EY','T','T','AH','T')	('W','W','IY','T','IY'),('AA','K'), ('T','AH'),('EY','T','T','AH','T')	WITTY ON THE OUTSIDE	5.6	14.3	25.0	0.0
EVEN BEFORE SHE ENTERED THE WATER	('IY','F','IY','K') ('P','IY','F','AO','W'), ('CH','IY'),('EY','K','T','ER','T'), ('T','AH'),('W','AO','T','ER')	('T','AH','IY','K'), ('P','IY','F','AO','W'), ('CH','IY'),('EY','K','T','ER','T'), ('AH'),('W','AO','T','ER')	DYING BEFORE SHE ENTERED A WATER	10.7	21.2	33.3	0.0
LIKE HUNDREDS OF THOUSANDS OF PEOPLE DO EVERY YEAR	('K','AH','K'), ('K','AH','K','T','W','AH','T','T'), ('AH','F'), ('T','EY','T','AH','K','T','T'), ('AH','F'), ('P','IY','P','AH','K'),('T','UH'), ('EY','F','ER','IY'),('K','IY','W')	('K','AH','K'), ('K','AH','K','T','W','AH','T','T'), ('AH','F'), ('T','EY','T','AH','K','T','T'), ('AH','F'), ('P','IY','P','AH','K','K','T','UH'), ('EY','F','ER','IY'),('K','IY','W')	NINE HUNDREDS OF THOUSANDS OF PEOPLES EVERY YEAR	2.2	10.0	33.3	0.0

V. CONCLUSION

A neural network-based lip reading system has been developed to predict sentences covering a wide range of vocabulary in silent videos from people speaking. The system is lexicon-free, uses only visual cues represented by visemes of a limited number of distinct lip movements, and is robust to different levels of lighting. Verified on the BBC LRS2 data set, the system has demonstrated a significant improvement on classification accuracy of words compared to the state-of-the-art works.

Future research includes investigating a more suitable neural network architecture in order to enable the system to have a good generalisation capability with a higher ratio of the number of training samples to the number of test samples.

In addition, an efficient conversion of visemes to words is crucial when using visemes as classification scheme for lip reading sentences. As shown in the experiments, although the classification accuracy of visemes achieved by the proposed system was very high (over 95%), the classification accuracy of words was significantly dropped after the conversion (65.5%). As such, it is important to explore any other possible approaches for the conversion. For perplexity analysis-based conversion, different global optimisation methods need to be considered while also limiting the computational overhead required.

ACKNOWLEDGMENT

This project was supported by a PhD Scholarship jointly-funded by Chinasoft International Ltd. and London South Bank University.

REFERENCES

[1] Z. Zhou, G. Zhao, X. Hong and M. Pietikinen. (2014). A review of recent advances in visual speech decoding. Image and vision computing.

[2] A. Fernandez-Lopez and F. Sukno. (2018). Survey on Automatic Lip-Reading in the Era of Deep Learning. Image and Vision Computing. 78.

[3] G. Potamianos, C. Neti, I. Matthews. (2004). Audio-visual automatic speech recognition: an overview, in: G. Bailly, E. Vatikiotis-Bateson, P. Perrier (Eds.). Issues in audio-visual speech processing, MIT Press.

[4] K. S. Talha et al. (2013). Speech Analysis Based On Image Information from Lip Movement. IOP Conference Series: Materials Science and Engineering.

[5] T. Stafylakis and G. Tzimiropoulos, (2017). Combining residual networks with LSTMs for lipreading. Proceedings of Interspeech.

[6] J. S. Chung, A. Zisserman, A. Senior and O. Vinyals. (2016). Lip Reading Sentences in the Wild. IEEE Conference on Computer Vision and Pattern Recognition.

[7] J. S. Chung and A. Zisserman. (2015). Lip Reading in the Wild. Asian Conference on Computer Vision.

[8] T. Afouras, J. S. Chung, A. Zisserman. (2018). Deep lip reading: a comparison of models and an online application. Proceedings of Interspeech.

[9] Y. M. Assael, B. Shillingford, S. Whiteson and N. de Freitas. (2016). LipNet: End-to-End sentence Level Lipreading. ICLR Conference.

[10] B. Shillingford, Y. Assael, M. Hoffman, T. Paine, C. Hughes, U. Prabhu, H. Liao, H. Sak, R. Hasim, H. Rao, L. Bennett, M. Mulville, B. Coppin, B. Laurie, A. Senior and N. Freitas. (2018). Large-Scale Visual Speech Recognition.

[11] A. J. Goldschen, O. N. Garcia and E. D. Petajan. (1996). Rationale for phoneme-viseme mapping and feature selection in visual speech recognition. Speechreading by Humans and Machines.

[12] P. F. Borwn et al. (1992). An Estimate of an Upper Bound for the Entropy of English. Computational Linguistics.

[13] Y. Fu, X. Zhou, M. Liu, M. Hasegawa-Johnson and T.S. Huang. (2007). Lipreading by locality discriminant graph. Proceedings of the International Conference on Image Processing.

[14] K. Kumar, T. Chen and R.M. Stern. (2007). Profile view lip reading. Proceedings of the International Conference on Acoustics, Speech and Signal Processing.

[15] P.J. Lucey, G. Potamianos and S. Sridharan. (2007). A unified approach to multi-pose audio-visual ASR, Proceedings of Interspeech.

[16] E. Marcheret, V. Libal and G. Potamianos. (2007). Dynamic stream weight modeling for audio-visual speech recognition. Proceedings of the International Conference on Acoustics, Speech and Signal Processing.

[17] S.J. Cox, R. Harvey, Y. Lan, J.L. Newman and B.J. Theobald. (2008). The challenge of multispeaker lip-reading. Proceedings of the International Conference on Auditory-Visual Speech Processing.

[18] I. Matthews, T.F. Cootes, J.A. Bangham, S. Cox and R. Harvey. (2002). Extraction of visual features for lipreading. IEEE Transactions in Pattern Analysis and Machine Intelligence.

[19] S.J. Cox, R. Harvey, Y. Lan, J.L. Newman and B. J. Theobald. (2008). The challenge of multi-speaker lipreading. Proceedings of the International Conference on Auditory-Visual Speech Processing.

[20] B. Lee, M. Hasegawa-Johnson, C. Goudeseune, S. Kamdar, S. Borys, M. Liu and T.S. Huang. (2004). AVICAR: audio-visual speech corpus in a car environment. Proceedings of Interspeech.

[21] H. Hofmann, S. Sakti, R. Isotani, H. Kawai, S. Nakamura and W. Minker. (2010). Improving spontaneous English ASR using a joint-sequence pronunciation model. 4th International Universal Communication Symposium, Beijing.

[22] V. I. Levenshtein. (1966). Binary codes capable of correcting deletions, insertions, and reversals. Soviet Physics Doklady.

[23] K. Saenko, K. Livescu, M. Siracusa, K. Wilson, J. Glass, and T. Darrell. (2005). Visual speech recognition with loosely synchronized feature streams. In Tenth IEEE International Conference on Computer Vision (ICCV).

[24] O. Koller, H. Ney, and R. Bowden. (2015). Deep learning of mouth shapes for sign language. IEEE International Conference on Computer Vision Workshop (ICCVW).

[25] A. B. Mattos, D. Oliveira, and E. Morais. (2018). Improving Viseme Recognition Using GAN-Based Frontal View Mapping. Analysis and Modeling of Faces and Gestures (CVPR).

[26] K. Thangthai, H. L. Bear, and R. Harvey. (2017). Comparing phonemes and visemes with dnn-based lipreading. In Proceedings of British Machine Vision Conference.

[27] S. Fenghour, D. Chen and P. Xiao. (2019). Decoder-Encoder LSTM for Lip Reading. Conference: 8th International Conference on Software and Information Engineering (ICSIE).

[28] Y. Bengio et al. (2009). Curriculum learning. Proceedings of the 26th annual international conference on machine learning.

[29] L. Elman. (1993). Learning and development in neural networks : the importance of starting small. In: 48, pp. 71-99.

[30] S. Lee and D. Yook. (2002). Audio-to-Visual Conversion Using Hidden Markov Models. In Proceedings of the 7th Pacific Rim International Conference on Artificial Intelligence: Trends in Artificial Intelligence.

[31] A. Botev, B. Zheng and D.Barber. (2017). Complementary sum sampling for likelihood approximation in large scale classification.

[32] J. Jeffers and M. Barley (1971). Speechreading (Lipreading). Charles C Thomas Publisher Limited.

[33] C. Neti et al. (2000). Audio visual speech recognition. Technical report IDIAP.

[34] T. J. Hazen, K. Saenko, C. La and J. R. Glass. (2004). A segment based audio-visual speech recognizer: data collection, development, and initial experiments. Proceedings of the 6th International Conference on Multimodal Interfaces.

[35] E. Bozkurt, C. E. Erdem, E. Erzin, T. Erdem and M. Ozkan. (2007). Comparison of phoneme and viseme based acoustic units for speech driven realistic lip animation. In 3DTV Conference.

[36] C. Fisher. (1968). Confusions among visually perceived consonants. Journal of Speech, Language, and Hearing Research.

[37] F. DeLand. (1931). The story of lip-reading, its genesis and development.

[38] W. B. Dolan and C. Brockett. (2005). Automatically constructing a corpus of sentential paraphrases. In Proceedings of the Third International Workshop on Paraphrasing.

[39] S. Gray, A. Radford, and K. P. Diederik. (2017). Gpu kernels for block-sparse weights.

[40] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. (2017). Attention Is All You Need. NIPS.
[41] R. Treiman, B. Kessler and S. Bick. (2001). Context sensitivity in the spelling of English vowels. Journal of Memory and Language.
[42] A. Radford, K. Narasimhan, T. Salimans and I. Sutskever. (2018). Improving Language Understanding by Generative Pre-Training.
[43] C.E. Shannon. (1948). A Mathematical Theory of Communication. The Bell System Technical Journal.
[44] D. P. Kingma and J. Ba. (2015). Adam: A method for stochastic optimization. Proceedings of ICLR.
[45] S. Fenghour, D. Chen, P. Xiao and K. Guo. (2020). Disentangling Homophemes in Lip Reading using Perplexity Analysis.
[46] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.Y. Fu and A.C. Berg. (2016). Ssd: Single shot multibox detector. Proceedings of ECCV. pp. 21-37. Springer.
[47] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou and M. Pantic. (2013). 300 faces in-the-wild challenge: The first facial landmark localization challenge. In IEEE International Conference on Computer Vision Workshops.
[48] S. Yang, Y. Zhang, D. Feng, M. Yang, C.Wang, J. Xiao, K. Long, S. Shan and X. Chen. (2019). LRW-1000: A naturally-distributed large-scale benchmark for lip reading in the wild. International Conference on Automatic Face and Gesture Recognition.
[49] M. Cooke, J. Barker, S. Cunningham and X. Shao. (2006). An audio-visual corpus for speech perception and automatic speech recognition, Journal Acoustics Society America 120 (5).
[50] I. Anina, Z. Zhou, G. Zhao and M. Pietikainen. (2015). OuluVS2: a multi-view audiovisual database for non-rigid mouth motion analysis. Proc. International Conference on Automatic Face and Gesture Recognition.
[51] J.S. Chung and A. Zisserman. (2017). Lip reading in profile. Proceedings of the British Machine Vision Conference.
[52] S. Petridis, Z. Li and M. Pantic. (2017). End-to-end visual speech recognition with LSTMs. Proceedings of the International Conference on Acoustics, Speech and Signal Processing.
[53] S. Petridis, Y. Wang, Z. Li and M. Pantic. (2017). End-to-end audiovisual fusion with LSTMs, Proc. International Conference on Auditory-Visual Speech Processing.
[54] S. Petridis, Y. Wang, Z. Li and M. Pantic,. (2017). End-to-end multi-view lipreading. Proceedings of the British Machine Vision Conference.
[55] H.L. Fung and B. Mak. (2018). End-to-end low-resource lip-reading with maxout CNN and LSTM, Proceedings of the International Conference on Acoustics, Speech and Signal Processing.
[56] S. Petridis, T. Stafylakis, P. Ma, F. Cai, G. Tzimiropoulos and M. Pantic. (2018). End-to-end audiovisual speech recognition. Proceedings of the International Conference on Acoustics, Speech and Signal Processing.
[57] S. Petridis, J. Shen, D. Cetin and M. Pantic. (2018). Visual-only recognition of normal, whispered and silent speech. Proceedings of the International Conference on Acoustics, Speech and Signal Processing.
[58] M. Wand, N.T. Vu and J. Schmidhuber. (2018). Investigations on end-to-end Audiovisual fusion. Proceedings of the International Conference on Acoustics, Speech and Signal Processing.
[59] K. Xu, D. Li, N. Cassimatis and X. Wang. (2018). LCANet: end-to-end lipreading with cascaded attention-CTC. Proceedings of the International Conference on Automatic Face and Gesture Recognition.
[60] C. Wang. (2019). Multi-grained spatio-temporal modelling for lip-reading. British Machine Vision Conference.
[61] X. Weng and K. Kitani. (2019). Learning spatio-temporal features with two-stream deep 3D CNNs for lip reading. British Machine Vision Conference.
[62] B. Martinez, P. Ma, S. Petridis and M. Pantic. (2020). Lipreading using Temporal Convolutional Networks.
[63] T. Afouras, J. S. Chung and A. Zisserman. (2018). LRS3-TED: a large-scale dataset for visual speech recognition.
[64] A. B. Mattos, D. Oliveira and E. Morais. (2018). Improving CNN-based Viseme Recognition Using Synthetic Data. 10.1109/ICME.2018.8486470.
[65] P. C. Hanavan. (1954). Audiovisual Speech Perception.

SOUHEIL FENGHOUR received an MSci in Physics from Imperial College, London, UK in 2012. From 2012 to 2016 he has worked in various internet companies as a Data Analyst doing data mining and analytics. He is currently pursuing a PhD degree in Computer Science at London South Bank University. His research interests include lip reading, deep learning, natural language processing, computer vision and heuristic search optimisation.

DAGING CHEN received his bachelor's degree in Systems Engineering in 1982 from Northwestern Polytechnical University, Xian, China, and his MPhil degree in Automatic Control Engineering in 1990 from the National University of Defense Technology, Changsha, China. He earned his PhD degree in Automatic Control Engineering in 1993 from Northwestern Polytechnical University, Xian, China. From 1994 to 1997, he worked as a Postdoctoral Researcher and then an Associate Professor at the National Key Laboratory of Radar Signal Processing, Xidian University, Xian, China. From 1997 to 1998, he was a Research Associate in the Department of Computer Science and Engineering, The Chinese University of Hong Kong. From 1998 to 1999, he worked as a Research Fellow in the System, Electronics and Information Laboratory, IRESTE, University of Nantes, Nantes, France. Since 1999 he has been working at London South Bank University, and currently is a Senior Lecturer in Informatics in the School of Engineering. His research interests include deep learning algorithms with applications in lip reading, medical image diagnosis, high dimensional data embedding and visualization, high-volume data labelling, and business intelligence.

KUN GUO received the Bachelor degree in Detection, Guidance and Control Technology in 2007 and the Masters degree in Systems Engineering in 2010 from Northwestern Polytechnical University, Xi'an, China. He received his PhD degree in Engineering Systems and Design in 2013 from London South Bank University, UK. From 2014 to 2016, he worked as a Senior Data Analyst at ZTE, Xi'an, China. Since 2016, he has been working at Xi'an VANXUM Electronics Technology Co., Ltd., China, as a Senior Algorithm Engineer. His research interests include applications of deep learning in computer vision and natural language processing, data mining and computer graphics compression.

PERRY XIAO received a Bachelors degree in Opto-electronics in 1990, a Masters degree in Solid State Physics in 1993 both from Jilin University of Technology, China; and the PhD in Photophysics from Strathclyde University and London South Bank University in 1998. From 1998 to 2000, he worked as a Research Fellow in the School of Engineering at London South Bank University and has held various posts at the university since 2000. He is also the co-founder and director of Biox Systems Ltd, a successful university spin-out company that designed and manufactured - AquaFlux and Epsilon, novel instruments for water vapour flux density and permittivity imaging measurements, which have been sold to more than 200 organizations worldwide, including leading cosmetic companies such as Unilever, L'Oreal, Philips, GSK, Johnson and Johnson, and Pfizer. His main research interest is to develop novel infrared and electronic measurement technologies for biomedical applications, including skin characterisation, trans-dermal drug diffusion and medical diagnosis.