Computers, Materials & Continua
DOI:10.32604/cmc.2021.016509
Article

Tech Science Press

HLR-Net: A Hybrid Lip-Reading Model Based on Deep Convolutional Neural Networks

Amany M. Sarhan[1], Nada M. Elshennawy[1] and Dina M. Ibrahim[1,2,*]

[1]Department of Computers and Control Engineering, Faculty of Engineering, Tanta University, Tanta, 37133, Egypt
[2]Department of Information Technology, College of Computer, Qassim University, Buraydah, 51452, Saudi Arabia
[*]Corresponding Author: Dina M. Ibrahim. Emails: d.hussein@qu.edu.sa, dina.mahmoud@f-eng.tanta.edu.eg
Received: 04 January 2021; Accepted: 17 February 2021

Abstract: Lip reading is typically regarded as visually interpreting the speaker's lip movements during the speaking. This is a task of decoding the text from the speaker's mouth movement. This paper proposes a lip-reading model that helps deaf people and persons with hearing problems to understand a speaker by capturing a video of the speaker and inputting it into the proposed model to obtain the corresponding subtitles. Using deep learning technologies makes it easier for users to extract a large number of different features, which can then be converted to probabilities of letters to obtain accurate results. Recently proposed methods for lip reading are based on sequence-to-sequence architectures that are designed for natural machine translation and audio speech recognition. However, in this paper, a deep convolutional neural network model called the hybrid lip-reading (HLR-Net) model is developed for lip reading from a video. The proposed model includes three stages, namely, preprocessing, encoder, and decoder stages, which produce the output subtitle. The inception, gradient, and bidirectional GRU layers are used to build the encoder, and the attention, fully-connected, activation function layers are used to build the decoder, which performs the connectionist temporal classification (CTC). In comparison with the three recent models, namely, the LipNet model, the lip-reading model with cascaded attention (LCANet), and attention-CTC (A-ACA) model, on the GRID *corpus* dataset, the proposed HLR-Net model can achieve significant improvements, achieving the CER of 4.9%, WER of 9.7%, and Bleu score of 92% in the case of unseen speakers, and the CER of 1.4%, WER of 3.3%, and Bleu score of 99% in the case of overlapped speakers.

Keywords: lip-reading; visual speech recognition; deep neural network; connectionist temporal classification

1 Introduction

Lip reading can be defined as the ability to understand what people are saying from their visual lip movement. Lip reading is a difficult task for humans because lip movements corresponding to different letters are visually very similar (e.g., b and p, or d and t) [1,2]. Automatic lip reading is currently used in many applications, either as a standalone application or as a

This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

supplementary one. It has been considered as a human-computer interaction approach that has been recently trending in the literature. One of the famous applications of lip reading is as an assisting tool for deaf persons by transforming the speech in the form of a video into subtitles. It can also be used as a supplementary means in noisy or virtual reality (VR) environments by completing the unheard sentences affected by the noise existence. In addition, it can greatly enhance the real experience of immersive VR [3]. Besides, automatic lip reading has been used in many applications, including security, speech recognition, and assisted driving systems. Automatic lip reading involves several tasks from different fields, including image processing, pattern recognition, computer vision, and natural language understanding.

Machine learning (ML) is a type of models that allows software applications to become more accurate in predicting outcomes without being explicitly programmed [4,5]. The basic premise of machine learning is to build algorithms that can predict output data based on the input data using statistical analysis and update output data when new input data become available. Deep learning is a machine learning method that uses an input X to predict an output Y. For instance, for given stock prices of the past week, a deep learning algorithm can predict the stock price of the next day [6]. Given a large number of input-output data pairs, a deep learning algorithm aims to minimize the difference between the predicted and expected outputs by learning the relationship between the input and output data, which enables the deep learning model to generalize accurate outputs for previously unseen inputs.

Lip reading is considered to be a difficult task for both humans and machines because of the high similarity of lip movements corresponding to uttering letters (e.g., letters b and p, or d and t). In addition, lip size, orientation, wrinkles around the mouth, and brightness also affect the quality of the detected words. These problems can be addressed by extracting the spatio-temporal features from a video and then mapping them to the corresponding language symbols, which represents a nontrivial learning task [7]. Due to its hardness, machine learning approaches have been proposed.

Recently, deep learning approaches have been applied to lip reading [7–9], which resulted in the fast development of the speech recognition field. In general, these approaches first extract the mouth area from a video of interest and then feed the extraction result to the input of a deep learning-based model. This model is commonly trained such that both feature extraction and word identification are automatically performed.

Currently, most of the existing lip-reading methods are based on the sequence-to-sequence architectures that are designed for applications such as natural language translation and audio speech recognition. The most common lip-reading methods are the LipNet model [7], lip-reading model with cascaded attention-CTC (LCANet) [10], which is also known as the attention high network-CTC (AH-CTC), and attention-CTC (A-CTC) models [6]. However, the lip-reading performance requires further improvement, which can be achieved by using more robust deep learning models and utilizing available datasets. The ambiguity of the translation from videos to words makes lip reading a challenging problem, which has not been solved yet. To address these problems, in this paper, a video-based sentence-to-sentence lip-reading model is developed using a deep convolutional neural network model, and it is denoted as the hybrid lip-reading network (HLR-Net). The proposed model consists of three stages: pre-processing, encoder, and decoder stages. The pre-processing stage is responsible for extracting the mouth movements from the video frames, frame normalization, and sentence preparation. The encoder is built of the inception, gradient, and bidirectional GRU layers, while the decoder consists of the attention, fully-connected, activation function layers, and connectionist temporal classification (CTC). The proposed model output is a subtitle of the input video provided in the form of a sentence.

The rest of this paper is organized as follows. Section 2 summarizes the recent lip-reading related work categorized from different perspectives. Section 3 describes the proposed model. Section 4 presents and discusses the experimental results. Finally, Section 5 draws the conclusion and introduces the ongoing future work.

2 Related Work

Introducing artificial intelligence to lip reading can greatly help deaf people by providing an automated method to understand video data presented to them. Millions of people around the world suffer from a certain extent of hearing deficiency, and developing a suitable lip-reading model can help them to understand other people' speech and thus allow them to participate in conversations, thus making them be connected to the real world. However, developing such a model is challenging for both designers and researchers. These models should be well designed, perfected, and integrated into smart devices to be widely available to all people in need of speech-understanding assistance.

Generally speaking, speech recognition can be conducted on the letter, word, sentence, digit, or phrase level. Also, it can be based on a video with or without a voice. Some of the recent studies on word-level lip reading have been focused on speaker-independent lip reading by adapting a system using Speaker Adaptive Training (SAT) technique, which was originally used in the speech recognition field [3]. The feature dimension was reduced to 40 using the linear discriminant analysis (LDA), and then the features were decorrelated using the maximum likelihood linear transform (MLLT). Namely, the 40-dimensional speaker-adapted features were spliced across a window of nine frames first, and then the LDA was applied to decorrelate the concatenated features and reduce the dimensionality to 25 Next, the obtained features were fed to the input of a context-dependent deep neural network (CD-DNN). In this way, the error rate of the speaker-independent lip reading was significantly reduced. Furthermore, it has been shown that the error can be even further reduced by using additional deep neural networks (DNNs). It has also been proven that there is no need to transform phonemes to visemes to apply the context-dependent visual speech transcription.

In [4], a method for automatically collecting and processing very large-scale visual speech recognition data using British television broadcasting was proposed. The proposed method could process thousands of hours of spoken text covering it into the data having an extensive vocabulary of thousands of different words. To validate the method, the VGG-M, 3D convolution with early fusion, 3D convolution with multiple towers, multiple towers, and early fusion models were used. The input image size was chosen to be 112×112 pixels. Multiple towers and early fusion models achieved the best accuracy among those models when testing in 500 and 333 classes. A learning architecture for word-level visual speech recognition was presented in [8]. This model combined spatiotemporal convolutional, residual (ResNet), and bidirectional LSTM networks. The ResNet building blocks were composed of two convolutional layers with BN and ReLU activation functions, while the skip connections facilitated information propagation in the max-pooling layers. This model ignored irrelevant parts of utterance and could detect target words without the knowledge about word boundaries. The database entries were fully automatic, and the words in subtitles were identified by using the optical character recognition (OCR) technique and synchronized the audio data. This model incorporated data augmentation processes, such as applying random cropping and horizontal flips, during training. The proposed model achieved a word recognition accuracy of 83%.

In [10], three types of visual features were studied from the image-based and model-based aspects for a professional lip-reading task. These features included the lips ROI, lip-shape geometrical representation, and deep bottle-neck features. A six-layer deep auto-encoder neural network (DANN) was used to extract the three mentioned features. These features were then used in two lip reading systems: the conventional GMM-HMM system and the DNN-HMM hybrid system. Based on the reported results, the DBNFs system achieved an average relative improvement of 15.4% compared to the shape features system, while the shape features system achieved an average relative improvement of 20.4% compared to the ROI features system when applied to test data.

In [11], the authors targeted the lip-reading problem using only video data and considered variable-length sequence frames words or phrases. They designed a twelve-layer convolutional neural network (CNN) using two batch-normalization layers to train the model and to extract the visual features in the end-to-end. The aim of using the batch normalization was to decrease the internal and external variances in the features that could affect speech-recognition performance, such as speaker's accent, lighting and quality of image frames, the pace of the speaker, and posture of speaking. To avoid the problem of a variable speaking speed of different speakers, a concatenated lip image was created by extending the sequence to a fixed length. The MIRACLE-VC1 dataset was used to evaluate the system, and a 96% training accuracy and a 52.9% validation accuracy were achieved.

The performances of speaker-dependent and speaker-independent lip-reading models based on CNNs, such as AlexNet, VGG, HANN, and Inception V3, have been studied in [12–15]. The main ideas and findings of the previous research on lip reading supported by AI methods, the type of used dataset, and achieved accuracy values are given in Tab. 1. As shown in Tab. 1, the speech-recognition performance was evaluated by using only one metric, which was the accuracy.

Table 1: Comparison between earlier work based on the accuracy as a performance metric

Ref.	AI method	Accuracy	Dataset		Task
			Name	Size	
[3]	MLLT + SAT, DNN	48% mean (visemes) 52% mean phonemes.	200 sentences selected from the RM *corpus*.	only the front view vocabulary size of around 1000 words	Word
[4]	VGG-M, 3D Conv. with Early Fusion and Multiple Towers	92.5% at sentence level 88.6% in unseen speakers	Their own dataset	29 speakers 118,166 Utterances Duration 33 h.	Sentences
[8]	Spatiotemporal conv., residual and bidirectional LSTM networks.	83.0% at word level	Videos extracted from BBC TV broadcasts	500-size target-words with 1.28 sec video excerpts	Words
[10]	6-layer Deep Auto-encoder NN (DANN) GMM-HMM and DNN-HMM hybrid	15.4% Compared to shape features 20.4% Compared to ROI features	CUAVE	digits (0 to 9) 36 speakers (19 males and 17 females) 80 isolated digits	Isolated and connected digits
[11]	12-layer CNN with 2 layers of batch normalization	96.5% on training set 52.9% on validation set.	MIRACL-VC1	3000 instances	Word or phrase
[12]	CNN models: AlexNet and Inception V3	Speaker dependent AlexNet 86.6%, Inception V3 64.6% speaker independent AlexNet 37.1% inception-V3.17.6%	Miracl-VC1	15 speakers, 1500 instances	Word

(Continued.)

Table 1: Continued

Ref.	AI method	Accuracy	Dataset		Task
			Name	Size	
[13]	Pre-trained deep learning architecture VGG Net	94.86% in training, 93.82% in validation and 60% in testing	MIRACL-VC1 dataset with some modifications	15 speakers, 1500 instances	Word
[16]	Deep 3D CNNs, two-stream	84.07%	LRW	number of target words = 500	Word
[15]	CNN + Hahn moments	59.23%, 93.72%, and 90.86% on AV-Letters, OuluVS2 and BBC LRW, respectively.	AV-Letters, OuluVS2 and BBC LRW		Letters, digits or words

Various datasets have been used in the development of the lip-reading methods, including LSR, LSR2, MV-LSR, BCC, and CMLR, as summarized in Tab. 2. In [9], an aligned training *corpus* containing profile faces was constructed by applying a multi-stage strategy on (the LRS dataset) called the MV-LRS. In [17], lip reading was considered as an open-world problem containing unrestrained NL sentences. Their model was trained on the LRS dataset, and it achieved better performance than other compared methods on several datasets. In [18], another public dataset called the LRS2-BBC was introduced, and it contained thousands of natural sentences acquired from British television. Some of the transformer models [12] obtained the best result of 50% when they were decoded with a language method, achieving an improvement of over 20% compared to the previous result of 70.4% obtained by the state-of-the-art models. A lip by speech (LIBS) model was proposed in [19], and it supported the lip reading with speech recognition. The extracted features could provide complementary and discriminant features that could be assigned to the lip movements. In [20,21], the authors simplified the training procedure, which allowed training the model in a single stage. A variable-length augmentation approach was used to generalize the models to the variations found on the sequence length.

One of the most commonly used datasets in the word-level lip-reading models is the GRID *corpus* dataset [16]. In [6], the authors presented the LipNet model that mapped a variable-length sequence of video frames to the text at the sentence level and used the spatiotemporal convolution, recurrent network, and connectionist temporal classification loss. The input, which was a sequence of frames, was passed to three layers of the spatiotemporal CNN (STCNN) first and then to a spatial max-pooling layer. The extracted features were up-sampled and managed by a bidirectional ong short-term memory (Bi-LSTM). The LSTM output was passed to a two-layer feedforward network and a SoftMax network. The model training was performed using the CTC on the GRID *corpus* dataset from which the videos for speaker 21 were omitted because they were missing; also, all empty and corrupted videos were removed. The LipNet achieved the CER, WER, and accuracy of 2.4%, 6.6%, and 93.4%, respectively, when it was trained and tested on the GRID *corpus* dataset, which was the sentence-level dataset. This model has the advantage that it does not require alignment.

Furthermore, in [8], a DNN model was built using the feedforward and LSTM networks. The training of the model was performed using the error gradient backpropagation algorithm.

The GRID *corpus* dataset was used for training and testing the model, and 19 speakers with 51 different words were chosen to classify. The frame-level alignments obtained word-level segmentation of a video, producing a training dataset that consisted of $6 \times 1000 = 6000$ single words per speaker. A 40×40-pixel window containing the mouth area was detected from each video frame using the Gaussian function, a threshold value for the center determination and scaling parameter. The experiments were, however, speaker-dependent, and the experimental data were randomly divided into training, validation, and test data, while classifiers were always taken from the same speaker. The results were averaged over 19 speakers, and the word-recognition accuracy of 79.6% was achieved.

Table 2: Performance comparison between different AI methods with different Lip-reading datasets

Ref.	AI method	Performance measures			Dataset
		Accuracy	CER	WER	
[9]	CNN Single Shot Detector (SSD)	OuluVS2 **91.1%**	MV-LRS 54.4%	MV-LRS 62.8%	MV-LRS 74,574 sentences OuluVS2 52 subjects uttering 10 phrases
[17]	LSTM encoder CNN based on the VGG-M model.	lip only = 54.9% A/V clean = 87.4% A/V noisy = 75.3%	lip only = 39.5%, A/V clean = **7.9%** A/V noisy = 17.6%	lip only = 50.2% A/V clean = **13.9%** A/V noisy = 27.6%	LRS about 100,000 natural sentences from British television.
[18]	spatiotemporal ResNet + STFT + transformer CTC	–	–	Video only = 48.3% A/V = 8.2%	LRS2-BBC Compared with LRS3-TED
[19]	RNN-LSTMs Fully conv. model	–	–	50%	LRW 489K samples (500 words) LRS2 sentences up to 100 characters from BBC videos
[20]	Multi-scale Temporal Convolutional Networks (TCN).	–	LRW = 85.30% LRW 1000 = 41.4%	–	LRW LRW1000
[21]	Attention-based sequence-to-sequence architecture.	CMLR 69.99% LRS2 = 41.91%	CMLR = 31.27% LRS2 45.53%	LRS2 = 65.29%	CMLR (100,000 sentences) LRS2 (45,000 spoken sentences from BBC TV)

In [16], the authors presented a lip-reading deep neural network that utilized the asynchronous spiking outputs of the dynamic vision sensor (DVS) and dynamic audio sensor (DAS). The event-based features produced from the spikes of the DVS and DAS were used for the classification process. The GRID visual-audio lip reading dataset was used for model testing. Networks were trained both separately and jointly on the two modalities. It was concluded that the single-modality networks that were trained separately on DAS and DVS spike frames achieved lower accuracy than the single-modality networks that were trained on both the audio MFCC features and the video frames. The recurrent neural network (RNN) using the MFCC features achieved an accuracy of approximately 98.41%. In addition, the audio inputs yielded better performance than the corresponding video inputs, achieving an accuracy of 84.27%, which was expected because the audio was more informative than the lip movement for this task. When the jointly trained

network was tested on only the DVS spike frames, an accuracy of 61.64% was achieved, which represented a significant increase in the accuracy compared to the single video modality network, whose accuracy was 38.26%.

In [22], the authors conducted a comprehensive study analyzing the performances of some of the recent visual speech recognition (VSR) models for different regions of the speaker's face. The considered regions included the mouth, the whole face, the upper face, and the cheeks. Their study targeted the word-level and sentence-level benchmarks with different characteristics. Two models were used, the 3D-ResNet18 model, which was the word-level model, and the LipNet model, which was the sentence-level model. The experiments were conducted using two fixed-size mouth cropping methods: the fixed bounding box coordinates and the mouth-centered crops. Different methods of determining ROI regions were also considered. In addition, the LRW, LRW-1000, and GRID datasets were used. They concluded that using the information from the extraoral facial regions could enhance the VSR performance compared to the case when the use lip region was used as the model input. Accuracy on the word level was 85.02% on the LRW dataset, 45.24% on the LRW-1000 dataset, while on the sentence level, WER was 2.9%, and CER was 1.2%. The state-of-the-art research that used the GRID *corpus* dataset is given in Tab. 3. Based on the recent studies, the GRID *corpus* dataset is chosen to be used in this study.

Table 3: State-of-the-art researches based on the GRID *corpus* dataset

Ref.	AI method	Accuracy	CER	WER	Dataset name	Dataset size	Task type
[6]	3 layers of STCNN, spatial max-pooling, Bi-LSTM	93.4%	2.4%	6.6%	GRID *corpus* (sentence level dataset)	34 speakers, 1000 sentences (28 h across 34000 sentences	Sentences
[7]	Feedforward 2 RNN-LSTM 128 units	79.6%	–	–	GRID *corpus*	19 speakers, 51 different words	Word
[14]	3D CNN, Bi-GRU	97.2%	1.3%	2.9%	GRID *corpus*	34 speakers, 1000 sentences (28 h across 34000 sentences	Word
[10]	LSTM	96.9%	1.9%	4.8%	GRID *corpus* And LRW	34 speakers, 1000 sentences (28 h across 34000 sentences	Word
[16]	150-GRU RNN for audio features 3-layered CNN and a single 80-unit GRU layer for video features	RNN for DAS = 83.8% CNN + RNN DVS = 38.26% DAS + DVS = 86.66%	–	–	GRID visual-audio lip reading dataset	1000 sentences spoken by each of 34 talkers (18 males, 16 female), total 51 different words	Word
[22]	3D-ResNet18 (word level) LipNet (sentence level). Used Cutout technique to detect regions in the face.	LRW = 85.02% LRW-1000 = 45.24%	1.2%	2.9%	LRW, LRW-1000 (word level) and GRID (sentence level)	34 speakers video recording, yielding 33000 utterance	Word and sentence

3 Proposed Model

In this paper, a deep convolutional neural network model for lip reading from a video is developed and denoted as the HLR-Net model. The proposed model is built using CNN model

followed by an attention layer and a CTC layer. The CTC is a type of neural network layer having an associated scoring function as an activation function. The structure of the proposed HLR-Net model is presented in Fig. 1. The proposed model consists of three stages. The first stage is the pre-processing stage, which processes the input video by executing certain operations on it, including mouth movement extraction from the frames of movements, frame normalization, and finally, sentence preparation to obtain the input for the deep learning model. The second and third stages are the encoder and decoder stages that produce the output subtitle.

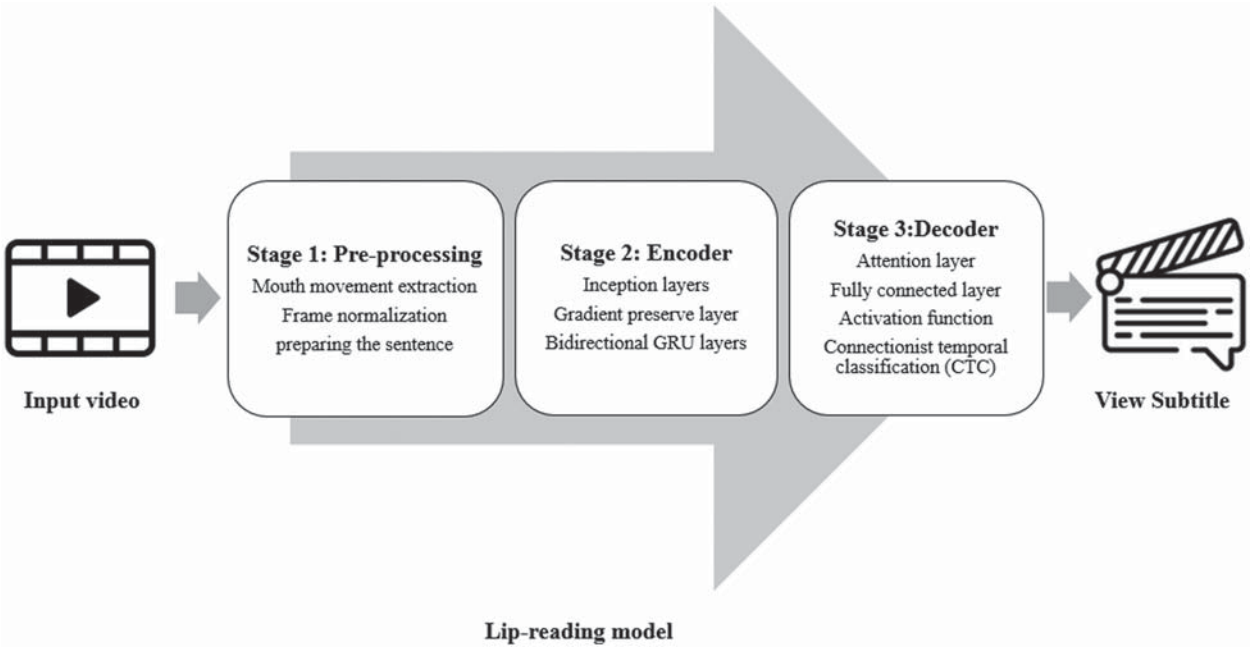

Figure 1: Abstraction view of our proposed HLR-Net model architecture

The second stage is composed of inception layers, gradient preservation layer, and bidirectional GRU layer, while the third stage consists of the attention layer, fully connected layer, activation function, and CTC layer. The proposed model is designed based on the attention deep learning model. It takes a video of lip movement as an input, then converts the video to frames using the OpenCV library, and extracts the mouth part using the dlip library. The resulting frames are normalized to obtain the final frame. The final frame is passed to the deep learning model to produce the final encoded sentence. In the next subsections, each stage is explained in more detail.

3.1 Stage 1: Data Pre-Processing

3.1.1 Mouth Movement Extraction

To perform mouth extraction, the dlib and OpenCV libraries are used to detect facial landmarks. Detection of facial landmarks can be considered as a shape prediction problem. An input image is fed to a shape predictor that attempts to localize points of interest regarding the shape, which are, in this case, the face parts as eyes, nose, and mouth. The facial landmark detection includes two steps, of which the first step is responsible for localizing the face in the image and detecting the key facial points, and the second step is responsible for detecting the mouth from

the face part, as shown in Fig. 2. As illustrated in Fig. 2, the mouth facial landmarks are located in between (49, 68) and represented by blue rectangles. Therefore, these points are extracted as a mouth in a frame with a size of 50 × 100, which is the final step of the mouth extraction.

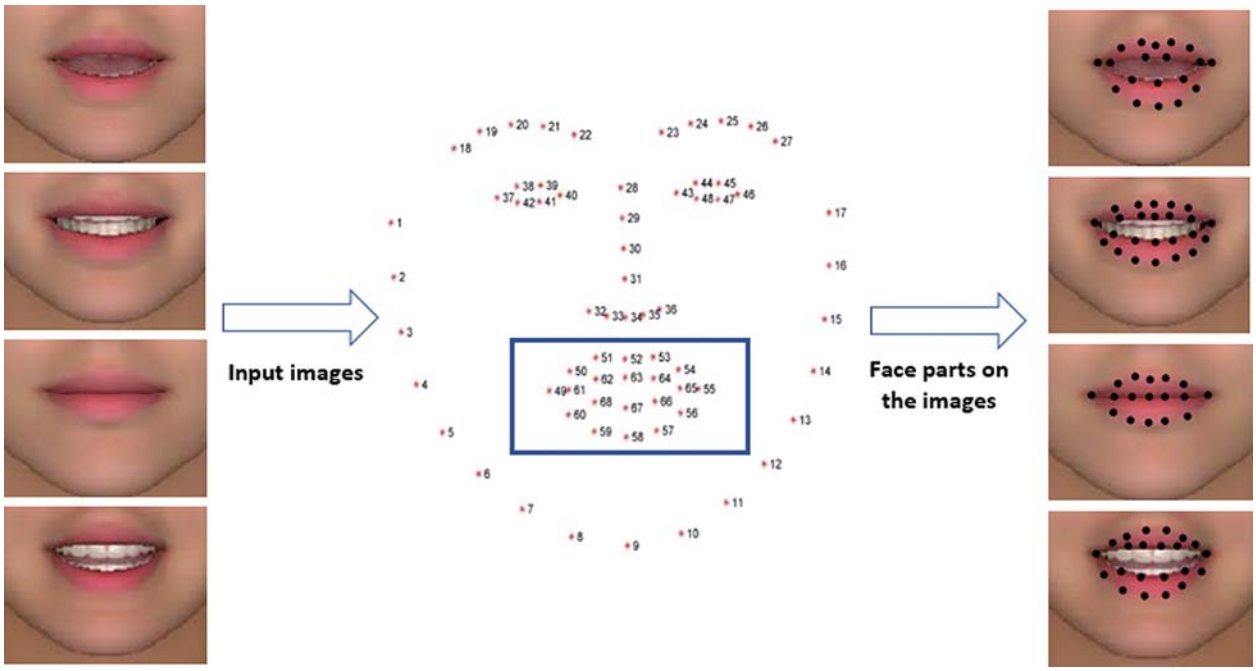

Figure 2: Face parts localization to the extracted images

3.1.2 Frame Normalization

The frame normalization process is to distribute the mouth pixels' values over the frame size. The video is divided into a number of frames for each mouth movement, and for each frame, the mouth localization and frame normalization are performed, as shown in Fig. 3.

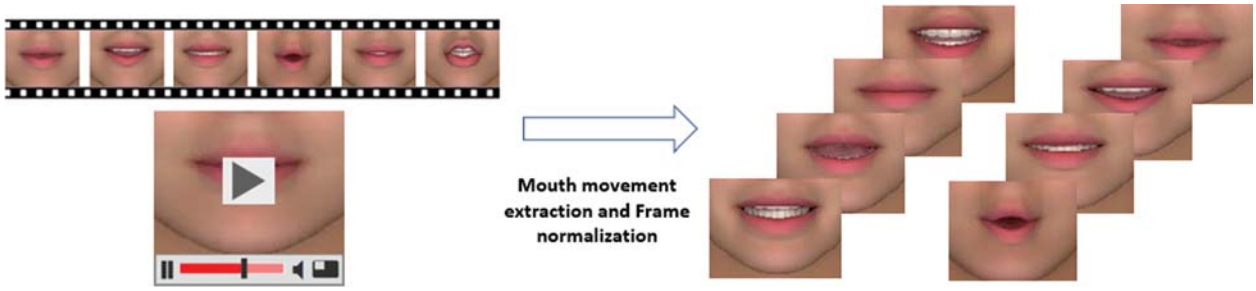

Figure 3: Mouth movement extraction and frame normalization from the input video

3.1.3 Sentence Preparation

For saving video sentences, the ".align" format is used. It is represented by a tuple for each word, represented by a frame, in the sentence. Each tuple contains the frame duration (start-time and end-time) in addition to the word. In order to separate adjacent sentences, the word "sil" is used. An example of a saved file is presented in Tab. 4. When a file is loaded, a sentence is constructed by appending the last column and ignoring "sil" and "sp" words. The sentence is then passed to a function that converts each sentence to a list of labels, where numbers refer to characters' orders in the sentence. In training, the sentence is passed together with the processed video to the video augmenter function, in which a label is assigned to each frame for training.

Table 4: Example of frame tuples for preparing a sentence

Frame start-time	Frame end-time	Word
0	23750	**sil**
23750	29500	bin
29500	34000	blue
34000	35500	at
35500	41000	f
41000	47250	two
47250	53000	now
53000	74500	**sil**

As previously mentioned, the input of the proposed model is a video, and its output is a mathematical representation of frames of the input video or in an array in the NumPy library. Thus, this stage can be summarized as follows: the input video is processed using the OpenCV library, and each frame of the video is captured. The number of frames per second (fps) is 25 fps, and the formed frames are normalized and converted to the form of a NumPy array. The NumPy array is fed directly to the proposed model.

3.2 Stage 2: Encoder Part

3.2.1 Three Inception Layers

Three layers of inception modules are used as CNNs to realize more efficient computation and deeper networks through dimensionality reduction with stacked $1 \times 1 \times 1$ convolutions. The modules are designed to solve the problem of computational cost, as well as overfitting, among other issues. The STCNNs are characterized by their processing time of video frames in the spatial domain.

The modules use multiple kernel filter sizes in the STCNN, but instead of stacking them sequentially, they are ordered to operate on the same level. By structuring the STCNN such that to perform the convolutions on the same level, the network becomes progressively wider but not deeper. To reduce the computationally cost even more, the neural network is designed such that an extra $1 \times 1 \times 1$ convolution is added before $3 \times 3 \times 3$ and $3 \times 5 \times 5$ layers. In this way, the number of input channels is limited, and $1 \times 1 \times 1$ convolutions are much cheaper than $3 \times 5 \times 5$ convolutions. The $1 \times 1 \times 1$ convolution layer is placed after the max-pooling layer. The most simplified version of an inception module works by performing a convolution on the input using three different sizes of filters ($1 \times 1 \times 1$, $3 \times 3 \times 3$, $3 \times 5 \times 5$) not only one. Likewise, max

pooling is performed. Then, the resulting outputs are concatenated and sent to the next layer. The details of the three stages of the proposed HLR-Net model are illustrated in Fig. 4.

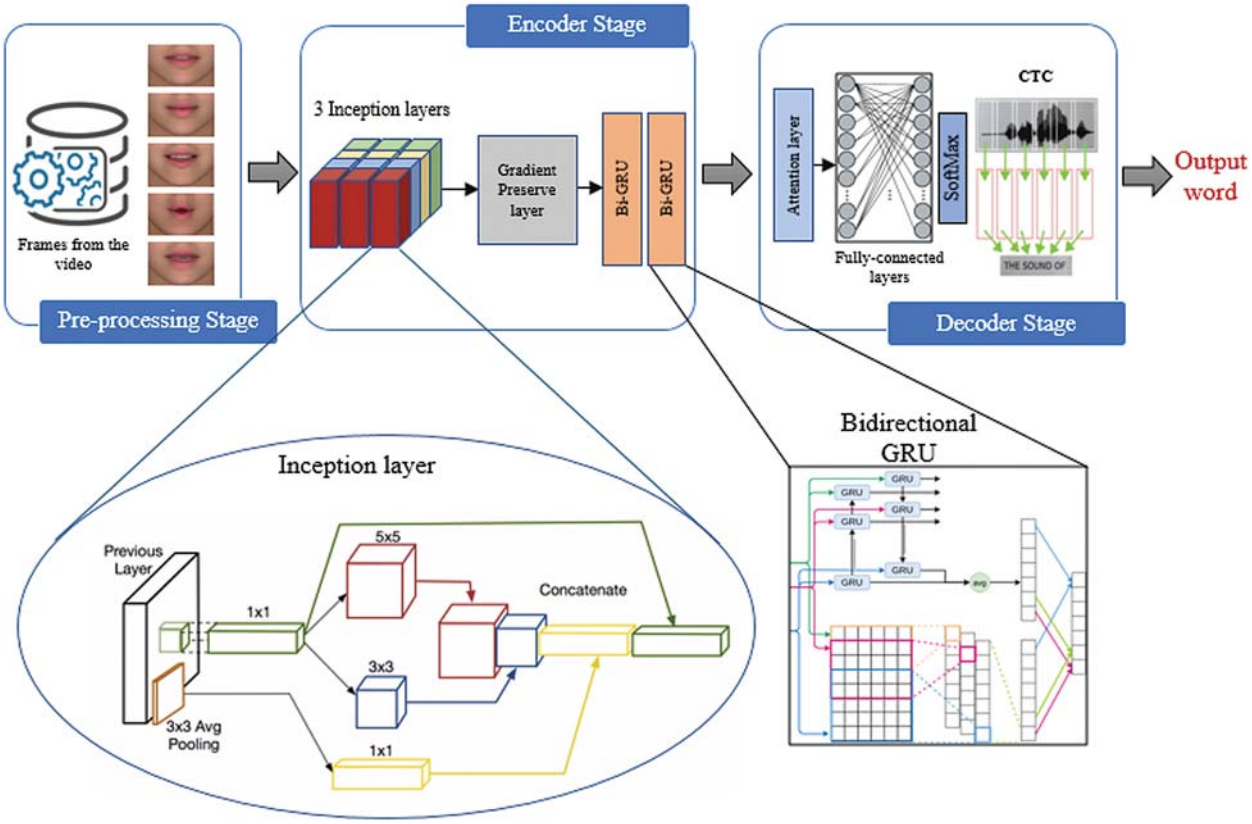

Figure 4: The proposed HLR-Net model architecture

3.2.2 Gradient Preservation Layer

The recurrent neural networks have a problem of gradient vanishing during the backpropagation. Namely, gradients are values used to update the neural network weights. The problem is that the gradient decreases what it back propagates through time, and if the gradient value becomes extremely small, it will slightly contribute to the learning process. For this reason, the gradient preservation layer is introduced. In this layer, two identity residual network blocks are used for solving the gradient vanishing problem. This layer is followed by a max-pooling layer to reduce the high-dimensionality problem.

3.2.3 Bidirectional GRU Layers

The output of the max-pooling layer is first flattened while preserving the time dimension and then passed to two bidirectional GRU (Bi-GRU) neural networks. The sequence model is used to add new information to spatial features. The bidirectional GRU can also be used with the attention model or a combination of the attention model and CTC. The Bi-GRUs denote improved versions of a standard RNN. To solve the gradient vanishing problem of a standard

RNN, GRU adopts the so-called update gate and reset gate. The reset gate performs the mixing of the current input and the previous memory state, and the updated gate specifies the portion taken from the previous memory state to the current hidden state. The Bi-GRU is used to capture both forward and backward information flows to attain the past and future states.

Basically, these are two vectors that decide what information should be passed to the output. It should be noted that they can be trained to keep information from a long time ago without vanishing it through time or removing information irrelevant to the prediction.

3.3 Stage 3: Decoder Part

3.3.1 Attention Layers

The output of the Bi-GRU in the attention layer is passed to the dense layer with 28 outputs representing the output characters. Attention layer is an example of a sequence-to-sequence sentence translation using a bidirectional RNN with attention. It represents the attention weights of the output vectors at each time step. There are several methods to compute the attention weights, for instance, by using the dot product or a neural network model with a single hidden layer. These weights are multiplied by each of the words in the source, and this product is fed to the language model along with the output from the previous layer to obtain the output for the current time step. These weights determine how much importance should be given to each word in the source to determine the output sentence. The final output is calculated using the SoftMax function in the dense layer.

3.3.2 CTC Layer

The CTC is an alignment-free, scalable, non-autoregressive method used in the sequence transduction in applications such as hand-written text recognition and speech recognition. The CTC is an independent component that is used to specify the output and scoring. It takes a group of samples sequence as an input and produces a label for each of them; it also generates blank outputs. When the number of observations is larger than the number of labels, the training is difficult; for instance, when there are multiple time slices that could correspond to a single phoneme. A probability distribution is used at each time step to predict the label as the alignment of the observed sequence is unknown [7,10]. The output of the CTC layer is continuous (for instance, obtained from a SoftMax layer), and it is adjusted and determined during the model training phase. The CTC deduce any recurrent characters that resolve to one character in spelling to form the right word. The CTC scores are processed by the backpropagation algorithm to update network weights. This approach makes the training process faster compared to that of RNN.

4 Experimental Results

4.1 Experimental Parameters and Datasets

The Google Colab with Pro version was used to test the proposed model. It had 2 TB storage with a server, which had 26 GB Ram and P100 GPU. There are many available lip-reading datasets, including AVICar, AVLetters, AVLetters2, BBC TV, CUAVE, OuluVS1, and OuluVS2 [15,23], but they are either plentiful, but include only single words, or are too small. In this study, the sentence-level lip reading was considered, so the Grid dataset [14] was used. This dataset included many audio and video recordings of 34 speakers, having 1000 sentences per speaker. In total, it consisted of 28-h data, including 34000 sentences.

For the HLR-Net model, the GRID *corpus* sentence-level dataset was used. The following simple structure of sentences was assumed: color (4) + command (4) + preposition (4) +

digit (10) + letter (25) + adverb (4). The number next to each part indicated the number of possible word choices for each of the six word groups. These groups consisted of the following data: {bin, lay, place, set}, {blue, green, red, white}, {at, by, in, with}, {A, . . . , Z}\{W}, {zero, . . . , nine}, and {again, now, please, soon}, and included a total of 64000 possible sentences. The code of the proposed HLR-Net has been uploaded to the GitHub website [24], and the parameters of the proposed HLR-Net are given in Tab. 5, where T refers to time, C refers to channels, F refers to feature dimension; H and W refer to height and width, respectively, and V refers to the number of words in the vocabulary, including the CTC blank symbol.

Table 5: The proposed HLR-Net architecture hyper parameters

Layer	Size/padding/stride/units	Input size	Dimension order
Incp STCNN	(1, 3, 5, maxp)/(1, 2, 2)/(1, 2, 2)/(32, 64, 16, 16)	75 × 50 × 100 × 3	T × H × W × C
Pool	(1, 2, 2)/(1, 2, 2)	75 × 52 × 27 × 12	T × H × W × C
Incp STCNN	(1, 3, 5, maxp)/(1, 1, 1)/(1, 1, 1)/(64, 128, 32, 32)	75 × 26 × 13 × 12	T × H × W × C
Pool	(1, 2, 2)/(1, 2, 2)	75 × 30 × 17 × 25	T × H × W × C
Incp STCNN	(1, 3, 5, maxp)/(1, 2, 2)/(1, 2, 2)/(96, 192, 48, 48)	75 × 15 × 8 × 256	T × H × W × C
Pool	(1, 2, 2)/(1, 2, 2)	75 × 12 × 6 × 384	T × H × W × C
Preserve	(1, 3, 1)/(1, 1, 1)/(1, 1, 1)/(96, 96, 384)	75 × 6 × 3 × 384	T × H × W × C
Pool	(1, 3, 3)/(1, 3, 3)	75 × 6 × 3 × 384	T × H × W × C
BI_GRU	256	75 × (2 × 1 × 384)	T × (H × W × C)
BI_GRU	256	75 × 512	T × F
Attention	28	75 × 512	T × F
Linear	28	75 × 28	T × F
SoftMax	–	75 × 28	T × W

Table 6: Mapping of 32 Phoneme to 12 viseme

Viseme	Phoneme
V1	/aa/ /ah/ /ay/ /eh/ /r/ /iy/
V2	/ae/ /ih/ /y/ /v/
V3	/aw/ /u/ /au/ /uw/
V4	/k/ /ch/
V5	/b/ /p/
V6	/d/ /t/
V7	/f/ /l/
V8	/ay/ /ih/
V9	/g/ /jh/
V10	/s/ /x/
V11	/m/ /n/
V12	/th/ /z/

To train the HLR-Net model, phoneme and viseme units were used. Visemes represented visually distinguishable speech units that had a one-to-many mapping to phonemes.

Phoneme-to-Viseme mapping [14] was used, and the mapping is shown in Tab. 6, where it can be seen that 32 phonemes were grouped into 12 visemes denoted by V1, V2, . . . , and V12.

The confusion matrix was used to test the visemes of the proposed model. The matrix showed that the proposed HLR-Net model could recognize elements with a little confusion. Namely, V1, V2, and V3, which mapped the phoneme groups {/aa/ /ah/ /ay/ /eh/ /r/ /iy/}, {/ae/ /ih/ /y/ /v/}, and {/aw/ /u/ /au/ /uw/}, respectively, were frequently misclassified during the text decoding process because they were similarly pronounced. However, misclassification of visemes was not signifivant, as shown in the confusion matrix in Fig. 5, which verified the effectiveness of the proposed HLR-Net model.

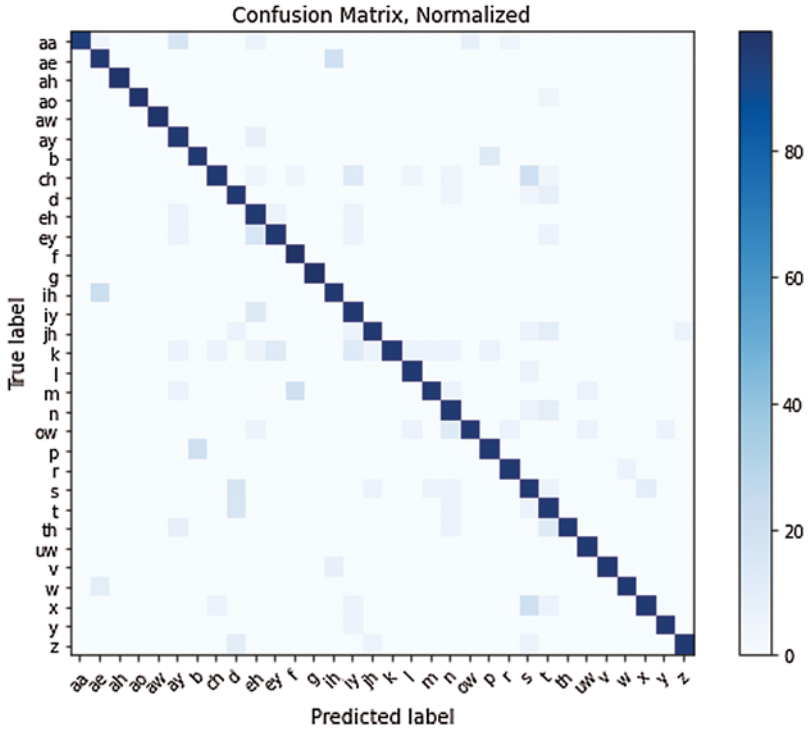

Figure 5: Confusion Matrix for the proposed HLR-Net model

4.2 Performance Metrics

To evaluate the performance of the proposed HLR-Net model and compare it with the baselines, the word error rate (WER), the character error rate (CER), and the bleu score [25,26] were used, and they were calculated by Eqs. (1)–(3), respectively.

$$WER = \frac{S+D+I}{N} = \frac{S+D+I}{S+D+C} \tag{1}$$

$$CER = \frac{i+s+d}{N} \tag{2}$$

$$Bleuscore = \frac{m}{wt} \tag{3}$$

In Eq. (1), S denotes the number of substitutions, D is the number of deletions, I is the number of insertions, C represents the number of correct words, and N is the number of words in the reference, and it is expressed as $N = S + D + C$. In Eq. (2), n is the total number of characters, and i denotes the minimal number of character insertions; s and d denote the numbers of substitutions and deletions required to transform the reference text into the output.

The Bleu score adopts a modified form of precision to compare a candidate translation against multiple reference translations. In Eq. (3), m denotes the number of words from the candidate, which are found in the reference, and wt is the total number of words of the candidate.

Table 7: Performance between our proposed HLR-Net proposed models and other recently work

Model	Unseen speakers			Overlapped speakers		
	CER (%)	WER (%)	Bleu score (%)	CER (%)	WER (%)	Bleu score (%)
Lipnet	6.4	11.4	88.2	1.9	4.8	96.9
LCANet (AH-CTC)	5.3	10.0	90.4	**1.3%**	2.9%	97.4
A-CTC	5.6	10.8	90.7	1.7%	4.1%	96.4
HLR-Net (Proposed)	**4.9**	**9.7**	**92**	1.4	**3.3**	**99**

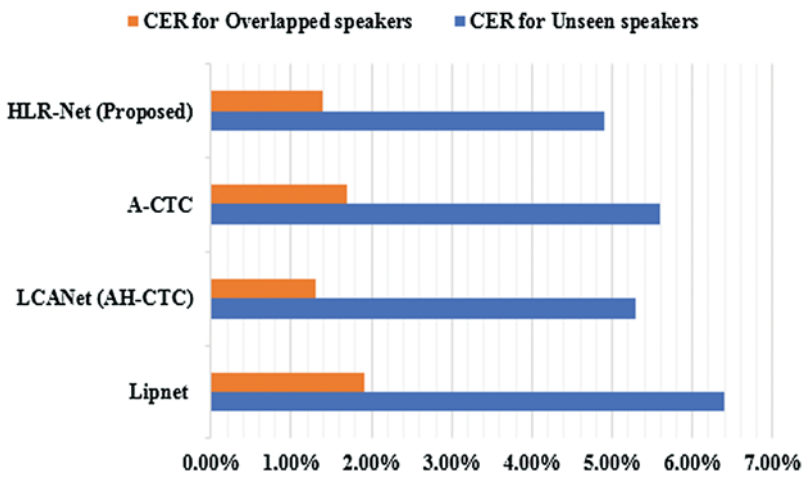

Figure 6: CER values for our proposed HLR-Net model compared with the other three model in case of unseen and overlapped speakers

4.3 Experimental Comparisons and Discussions

The CER/WER was defined as the least number of character (or word) substitutions, insertions, and deletions required to convert the prediction into the base truth, divided by the number of characters (or words) in the base. The Bleu score value indicated how similar the candidate text was to the reference texts, where values closer to 100% represented higher similarity. Smaller WER/CER values meant higher prediction accuracy, while a larger Bleu score was preferred. The results are given in Tab. 7. The proposed HLR-Net was compared with three recent

models: LipNet, LCANet (AH-CTC), and A-CTC models. All models were trained on the GRID dataset. The performances of the models were tested for two different types of speakers: unseen speakers and overlapped speakers. The comparisons of the models regarding the CER, WER, and Bleu score values are presented in Figs. 6–8, respectively.

In the case of unseen speakers, the proposed HLR-Net model achieved CER of 4.9%, WER of 9.7%, and Bleu score of 92%; the LipNet model achieved CER of 6.4%, WER of 11.4%, and Bleu score of 88.2%; the LCANet model achieved CER of 5.3%, WER of 10.0%, and Bleu score of 90.4%; lastly, the A-CTC model achieved CER of 5.6%, WER of 10.8%, and Bleu score of 90.7%.

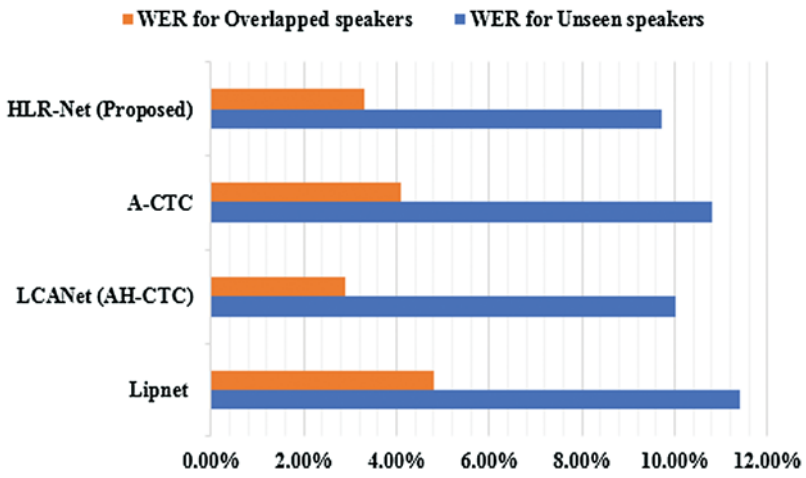

Figure 7: WER values for our proposed HLR-Net model compared with the other three model in case of unseen and overlapped speakers

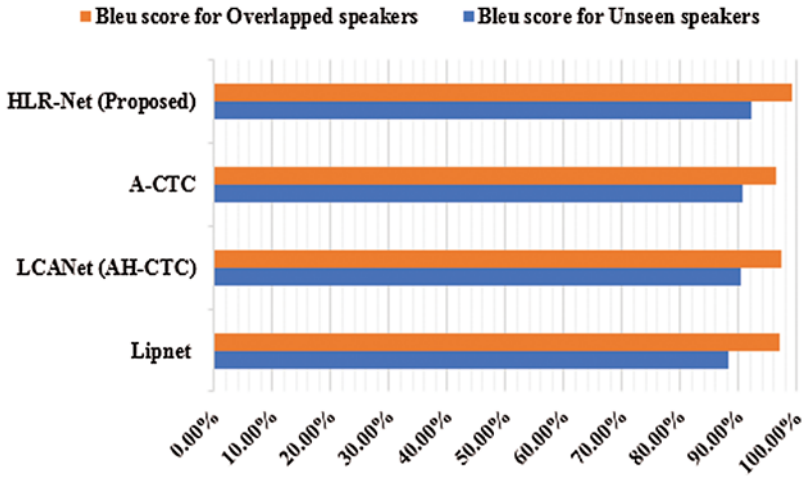

Figure 8: Bleu score values for our proposed HLR-Net model compared with the other three model in case of unseen and overlapped speakers

In the case of overlapped speakers, the proposed model also achieved better performance than the other models, having the CER of 1.4%, WER of 3.3%, and Bleu score of 99%. However, it should be noted that the CER value of the LCANet, which was 1.3%, was slightly better than that of the proposed model. Based on the overall result, the proposed HLR-Net model outperformed the other models.

5 Conclusions and Future Work

In this paper, a hybrid video-based lip-reading model is developed using deep convolutional neural networks and denoted as the HLR-Net model. The proposed model consists of three stages: pre-processing stage, encoder stage, and decoder stage. The encoder stage is composed of inception layers, gradient preservation layer, and bidirectional GRU layer, while the decoder consists of the attention layer, fully-connected layer, activation function, and CTC layer. The proposed model is designed based on the attention deep learning model. It uses a video of lip movement as an input, then converts this video to frames using the OpenCV library, and finally extracts the mouth part using the dlip library. The resulting frames are normalized to obtain the final frame. The final frame is passed to the deep learning model to produce the encoded sentence.

The performance of the proposed HLR-Net model is verified by the experiments and compared with those of the three state-of-the-art models, LipNet model, LCANet (AH-CTC) model, and A-CTC model. The experimental results show that the proposed HLR-Net model outperforms the other three models, achieving CER of 4.9%, WER of 9.7%, and Bleu score of 92% in case of unseen speakers, and CER of 1.4%, WER of 3.3%, and Bleu score of 99% in case of overlapped speakers. However, the CER value of the LCANet is 1.3%, and it is slightly better than that of the proposed model.

As future work, the proposed HLR-Net model will be applied and tested using the Arabic language for the purpose of further verification.

Acknowledgement: The authors would like to acknowledge the student group: Abdelrhman Khater, Ahmed Sheha, Mohamed Abu Gabal, Mohamed Abdelsalam, Mohamed Farghaly, Mohamed Gnana, and Hagar Metwalli from the Computer and Control Engineering Department at the Faculty of Engineering, Tanta University, who worked on this research as a part of their graduation project. We also thank LetPub (www.letpub.com) for its linguistic assistance during the preparation of this manuscript.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] Y. Zhang, S. Yang, J. Xiao, S. Shan and X. Chen, "Can we read speech beyond the lips? rethinking ROI selection for deep visual speech recognition," in *15th IEEE Int. Conf. on Automatic Face and Gesture Recognition (FG 2020)*, Buenos Aires, Argentina, pp. 356–363, 2020.

[2] Y. Zhao, R. Xu, X. Wang, P. Hou, H. Tang *et al.,* "Hearing lips: Improving lip reading by distilling speech recognizers," *Proc. of the AAAI Conf. on Artificial Intelligence*, vol. 34, no. 4, pp. 6917–6924, 2020.

[3] I. Almajai, S. Cox, R. Harvey and Y. Lan, "Improved speaker independent lip-reading using speaker adaptive training and deep neural networks," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Shanghai, China, pp. 2722–2726, 2016.

[4] J. S. Chung and A. Zisserman, "Lip reading in the wild," in *Asian Conf. on Computer Vision*, Berlin, Germany: Springer, pp. 87–103, 2016.
[5] B. Martinez, P. Ma, S. Petridis and M. Pantic, "Lipreading using temporal convolutional networks," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Barcelona, Spain, pp. 6319–6323, 2020.
[6] Y. Li, Y. Takashima, T. Takiguchi and Y. Ariki, "Lip reading using a dynamic feature of lip images and convolutional neural networks," in *IEEE/ACIS 15th Int. Conf. on Computer and Information Science*, Okayama, Japan, pp. 1–6, 2016.
[7] Y. M. Assael, B. Shillingford, S. Whiteson and N. de Freitas, "Lipnet: Sentence-level lipreading, vol. 2, no. 4, arXiv preprint arXiv: 1611.01599, 2016.
[8] M. Wand, J. Koutník and J. Schmidhuber, "Lipreading with long short-term memory," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Shanghai, China, pp. 6115–6119, 2016.
[9] T. Stafylakis and G. Tzimiropoulos, "Combining residual networks with LSTMs for lipreading," arXiv preprint arXiv: 1703.04105, 2017.
[10] J. S. Chung, A. Senior, O. Vinyals and A. Zisserman, "Lip reading sentences in the wild," in *IEEE Conf. on Computer Vision and Pattern Recognition*, Wuhan, China, pp. 3444–3453, 2017.
[11] F. Vakhshiteh and F. Almasganj, "Lip-reading via deep neural network using appearance-based visual features," in *24th National and 2nd IEEE Int. Iranian Conf. on Biomedical Engineering*, Suzhou, China, pp. 1–6, 2017.
[12] S. NadeemHashmi, H. Gupta, D. Mittal, K. Kumar, A. Nanda *et al.,* "A lip reading model using CNN with batch normalization," in *11th IEEE Int. Conf. on Contemporary Computing*, Aqaba, Jordan, pp. 1–6, 2018.
[13] I. Fung and B. Mak, "End-to-end low-resource lip-reading with Maxout CNN and LSTM," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Calgary, Alberta, Canada, pp. 2511–2515, 2018.
[14] K. Xu, D. Li, N. Cassimatis and X. Wang, "LCANet: End-to-end lipreading with cascaded attention-CTC," in *13th IEEE Int. Conf. on Automatic Face & Gesture Recognition*, Xi'an, China, pp. 548–555, 2018.
[15] T. Afouras, J. S. Chung and A. Zisserman, "LRS3-Ted: A large-scale dataset for visual speech recognition," arXiv preprint arXiv: 1809.00496, 2018.
[16] Z. Thabet, A. Nabih, K. Azmi, Y. Samy, G. Khoriba *et al.,* "Lipreading using a comparative machine learning approach," in *First IEEE Int. Workshop on Deep and Representation Learning*, Singapore, pp. 19–25, 2018.
[17] M. H. Rahmani and F. Almasganj, "Lip-reading via a DNN-HMM hybrid system using combination of the image-based and model-based features," in *3rd IEEE Int. Conf. on Pattern Recognition and Image Analysis*, Venice, Italy, pp. 195–199, 2017.
[18] P. Sindhura, S. Preethi and K. B. Niranjana, "Convolutional neural networks for predicting words: A lip-reading system," in *IEEE Int. Conf. on Electrical, Electronics, Communication, Computer, and Optimization Techniques*, Mysore, India, pp. 929–933, 2018.
[19] T. Afouras, J. S. Chung, A. Senior, O. Vinyals and A. Zisserman, "Deep audio-visual speech recognition," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, United States, 2018.
[20] X. Weng and K. Kitani, "Learning spatio-temporal features with two-stream deep 3D CNNs for lipreading," arXiv preprint arXiv: 1905.02540, 2019.
[21] M. A. Abrar, A. N. Islam, M. M. Hassan, M. T. Islam, C. Shahnaz *et al.,* "Deep lip reading-a deep learning based lip-reading software for the hearing impaired," in *IEEE R10 Humanitarian Technology Conf. (R10-HTC) (47129)*, Depok, Indonesia, pp. 40–44, 2019.
[22] A. H. Kulkarni and D. Kirange, "Artificial intelligence: A survey on lipreading techniques," in *10th IEEE Int. Conf. on Computing, Communication and Networking Technologies*, Kanpur, India, pp. 1–5, 2019.
[23] T. Shirakata and T. Saitoh, "Lip reading experiments for multiple databases using conventional method," in *58th Annual Conf. of the Society of Instrument and Control Engineers of Japan*, Hiroshima, Japan, pp. 409–414, 2019.

[24] GitHub, "A hybrid lip reading model based on deep CNNs," accessed 24 January 2021, 2020. [Online]. Available: https://github.com/Dr-Dina-M-Ibrahim/HLR-NET-A-Hybrid-Lip-Reading-model-based-on-Deep-Convolution-Neural-Networks.git.

[25] X. Li, D. Neil, T. Delbruck and S. C. Liu, "Lip reading deep network exploiting multi-modal spiking visual and auditory sensors," in *IEEE Int. Symp. on Circuits and Systems*, Sapporo, Hokkaido, Japan, pp. 1–5, 2019.

[26] A. Mesbah, A. Berrahou, H. Hammouchi, H. Berbia, H. Qjidaa *et al.,* "Lip reading with HAHN convolutional neural networks," *Image and Vision Computing*, vol. 88, no. 1, pp. 76–83, 2019.