LIP READING AND PREDICTION SYSTEM BASED ON DEEP LEARNING

Gishma K.M
Assistant professor of Computer Science and Engineering
Universal Engineering College
Kerala, India
geeshma123@gmail.com

Annmaria K.B
Department of Computer Science and Engineering
Universal Engineering College
Kerala , India
annmariakb2001@gmail.com

Ramna Parvan V.N
Department of Computer Science and Engineering
Universal Engineering College
Kerala, India
ramnanaazar111@gmail.com

Anagha Suresh
Department of Computer Science and Engineering
Universal Engineering College
Kerala , India
anaghasuresh2580@gmail.com

Athira Shaji
Department of Computer Science and Engineering
Universal Engineering College
Kerala , India
athirashajicpni@gmail.com

Abstract— **Speech perception is characterized as a multimodal process, which means it elicits several meanings. Understanding a message can be aided by, and in some cases even made necessary by, lip reading, which overlays visual cues on top of auditory signals. Lip-reading is a crucial field with many uses, including biometrics, speech recognition in noisy environments, silent dictation, and enhanced hearing aids. It is a challenging research project in the area of computer vision, whose major goal is to watch the movement of human lips in a video and recognize the textual content that goes with it. Yet, due to the constraints of lip changes and the depth of linguistic information, the complexity of lip identification has increased, which has slowed the growth of study themes in lip language. Nowadays, deep learning has advanced in several sectors, giving us the confidence to perform the task of lip recognition. Lip learning based on deep learning often entails extracting features and comprehending images using a network model, as opposed to classical lip recognition that recognizes lip characteristics. The design of the network framework for data gathering, processing, and data recognition for lip reading is the main topic of this discussion. In this research, we created a reliable and accurate method for lip reading. We first isolate the mouth region and segment it, after which we extract various aspects from the lip image, such as the Hog, Surf, and Haar features. Lastly, we use Gated Recurrent Units to train our deep learning model (GRU).**

Keywords—Haar,Hog and Surf features,GRU based deep learning Architecture

I. INTRODUCTION

When regular sound is not available, lip reading, often referred to as speechreading, is a method of understanding speech by visually analysing the movements of the lips, face, and tongue. Additionally, it depends on the context, linguistic proficiency, and any lingering hearing to deliver information. The majority of people with normal hearing process some speech information from sight of the moving mouth, even though lip reading is most commonly employed by the deaf and hard of hearing. In our system, we employ deep learning to separate speech from lip movements. Machine learning, which is simply a neural network with three or more layers, is a subset of deep learning. These neural networks make an effort to mimic how the human brain functions, however they fall far short of being able to match it, enabling it to "learn" from vast volumes of data. Additional hidden layers can help to tune and refine for accuracy even if a neural network with only one layer can still make approximation predictions. Reading human lips is a difficult

DOI:10.5281/zenodo.7993683

task. To predict spoken words, one needs to be aware of the underlying language as well as visual cues. To decode spoken words, experts need a particular amount of experience and awareness of visual expressions. Deep learning technology makes it possible to transform lip movements into meaningful speech today. With the aid of visual information, speech recognition in noisy circumstances can be improved.

II.LITERATURE REVIEW

Visual speech information plays an important role in automatic speech recognition (ASR) especially when audio is corrupted or even inaccessible. Despite the success of audio-based ASR, the problem of visual speech decoding remains widely open. This paper provides a detailed review of recent advances in this research area. In comparison with the previous survey [97] which covers the whole ASR system that uses visual speech information, we focus on the important questions asked by researchers and summarize the recent studies that attempt to answer them. In particular, there are three questions related to the extraction of visual features, concerning speaker dependency, pose variation and temporal information, respectively. Another question is about audio-visual speech fusion, considering the dynamic changes of modality reliabilities encountered in practice. In addition, the state-of-the-art on facial landmark localization is briefly introduced in this paper. Those advanced techniques can be used to improve the region-of-interest detection, but have been largely ignored when building a visual-based ASR system. We also provide details of audio-visual speech databases. Finally, we discuss the remaining challenges and offer our insights into the future research on visual speech decoding.

It combined three sub-networks: (i) The front-end, which applies spatiotemporal convolution to the frame sequence, (ii) a Residual Network(ResNet) that is applied to each time step, and (iii) the back end, which is a two-layer Bidirectional Long Short-Term Memory (Bi-LSTM) network. The SoftMax layer is applied to all time steps and the overall loss is the aggregation of the per time step losses, and the system is trained in an end-to-end fashion.Finally, the system performs not merely word recognition but also implicit key-word spotting, since the target words are not isolated, but they are part of whole utterances of fixed duration (1.28sec).

The model consists of multiple identical streams, one for each view, which extract features directly from different poses of mouth images. The temporal dynamics in each stream/view are modelled by a BLSTM and the fusion of multiple streams/viewstakes place via another BLSTM.

we propose a Temporal Focal block to sufficiently describe short-range dependencies and a Spatio-Temporal Fusion Module (STFM) to maintain the local spatial information and to reduce the feature dimensions as well. Fromthe experiment results, it is demonstrated that our method achieves comparable performance with the state-of-the-art approach using much less training data and much lighter Convolutional Feature Extractor. The training time is reduced by 12 days due to the convolutional structure and the local self-attention mechanism.

In this paper, we pro- pose a 3D CNNs based on ResNets toward a better action representation. We describe the training procedure of our 3D ResNets in details. We experimentally evaluate the 3D ResNets on the ActivityNet and Kinetics datasets. The 3D ResNets trained on the Kinetics did not suffer from overfit- ting despite the large number of parameters of the model, and achieved better performance than relatively shallow networks, such as C3D.

To verify the generalizability of the proposed method, we then fine-tune the pre-trained model on domain-specific datasets (GRID and TCD-TIMIT) for English speech reconstruction and achieve a significant improvement on speech quality and intelligibility compared to previous approaches in speaker-dependent and speaker-independent settings. In addition to English, we conduct Chinese speech reconstruction on the Chinese Mandarin Lip Reading (CMLR) dataset to verify the impact on transferability. Finally, we train the cascaded lip reading (video-to-text) system by fine-tuning the generated audios on a pre-trained speech recognition system and achieve the state- of-the-art performance on both English and Chinese benchmark datasets.

A thorough experimental evaluation on two large-scale lip reading benchmarks is presented with detailed analysis. The results accord with our motivation, and show that our method achieves state-of-the-art or comparable performance on these two challenging datasets.

This model is able to reach 94.2% of accuracy in the DMCLR dataset. Such performance makes it possible for Mandarin lip reading applications to be practical in real life. Additionally, we are able to achieve 86.6% and 57.2% accuracy on Lip Reading in the Wild (LRW) and LRW-1000 (Mandarin), respectively. The results show that our method achieves state-of- the-art performance on these two challenging datasets.

In order to improve the performance of machine lip reading, we propose a lip reading method based on 3D convolutional vision transformer (3DCvT), which combines vision transformer and 3D convolution to extract the spatio-temporal feature of continuous images, and take full advantage of the properties of convolutions and transformers to extract local and global features from continuous images effectively. The extracted features are then sent to a Bidirectional Gated Recurrent Unit (BiGRU) for sequence modeling. We proved the effectiveness of our method on large-scale lip reading datasets LRW and LRW-1000 and achieved state-of-the-art performance.

Motivated by this observation, we present LipNet, a model that maps a variable-length sequence of video frames to text, making use of spatiotemporal convolutions, a recurrent network, and the connectionist tempo- ral classification loss, trained entirely end-to-end. To the best of our knowledge, LipNet is the first end-to-end sentence-level lipreading model that simultaneouslylearns spatiotemporal visual features and a sequence model. On the GRID corpus,LipNet achieves 95.2% accuracy in sentence-level, overlapped speaker split task, outperforming experienced human lipreaders and the previous 86.4% word-level state-of-the-art accuracy (Gergen et al., 2016).

This work presents a scalable solution to open-vocabulary visual speech recognition. To achieve this, we constructed the largest existing visual speech recognition dataset, consisting of pairs of text and video clips of faces speaking (3,886 hours of video). In tandem, we designed and trained an integrated lipreading system, consisting of a video processing pipeline that maps raw video to stable videos of lips and sequences of phonemes, a scalable deep neural network that maps the lip videos to sequences of phoneme distributions, and a production-level speech decoder that outputs sequences of words.

We complete our visual speech modeling via hybrid DNN-HMMs and our visual speech decoder is a Weighted Finite-State Transducer (WFST). We use DCT and Eigenlips as a representation of mouth ROI image. The phoneme lipreading system word accuracy outperforms the viseme based system word accuracy. However, the phoneme system achieved lower accuracy at the unit level which shows the importance of the dictionary for decoding classification outputs into words.

We evaluate our method using the GRID cor- pus, which was processed to extract viseme images and their corresponding synthetic frontal views to be further classi- fied by our CNN model. Our results demonstrate that the additional synthetic frontal view is able to improve accu- racy in 5.9% when compared with classification using the original image only.

We show that this reliance on CNNs is not necessary and a pure transformer applied directly to sequences of image patches can perform very well on image classification tasks. When pre-trained on large amounts of data and transferred to multiple mid-sized or small image recognition benchmarks (ImageNet, CIFAR-100, VTAB, etc.), Vision Transformer (ViT) attains excellent results compared to state-of-the-art convolutional networks while requiring sub- stantially fewer computational resources to train.

Through the use of a Long-Short Term Memory Network with word embeddings, we can distinguish between homopheme words or words that produce identical lip movements. The neural network architecture achieved a character accuracy rate of 77.1% and a word accuracy rate of 72.2%.

To make the learning process more targeted at each particular language, an extra task of predicting the language identity is introduced in the learning process. Finally, a thorough comparison on LRW (English) and LRW-1000 (Mandarin) is performed, which shows the promising benefits from the synergized learning of different languages and also reports a new state-of-the-art result on both datasets.

Next, a bidirectional ConvLSTM augmented with temporal attention aggregates spatio-temporal information in the entire input sequence, which is expected to be able to capture the coarse-gained patterns of each word and robust to various conditions in speaker identity, lighting conditions, and so on. By making full use of the information from different lev- els in a unified framework, the model is not only able to distinguish words with similar pronunciations, but also becomes robust to appearance changes. We evaluate our method on two challenging word-level lip-reading benchmarks and show the effectiveness of the proposed method, which also demonstrate the above claims.

In this paper, a neural network-based lip reading system is proposed. 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.

To the best of our knowledge, this is the first end-to-end low-resource lip-reading system that does not require any separate feature extraction stage nor pre-training phase with external data re- sources. This is also the first work that utilizes maxout units in both CNN and LSTM in one single deep neural network.

III.EXISTING SYSTEM

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.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 recognize. 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.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 for classifying visemes, and a conversion of recognized visemes to possible words using perplexity analysis.

IV.PROPOSED SYSTEM

The main focus of the project is on deploying efficient and accurate deep learning models by utilizing the HOG, HAAR, and SURF algorithms in order to makes faster classification and the whole system can be implemented in a cost-effective way.

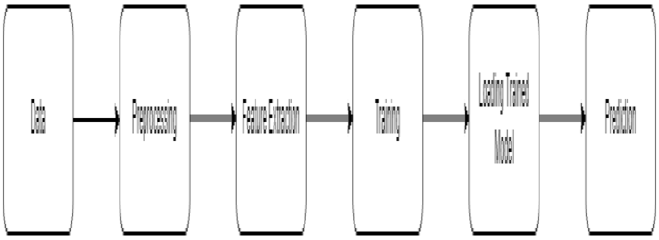


Figure 1 Block Diagram

The system as a whole consists of about 4 components. The system's first module is the dataset import and preprocessing module. Preparing raw data to be appropriate for a machine learning model is known as data preprocessing. In order to build a machine learning model, it is the first and most important stage. Real-world data typically includes noise, missing values, and may be in a useless format, making it impossible to use machine learning models on it directly. Data preprocessing is necessary to clean the data and prepare it for a machine learning model, which also improves the model's precision and effectiveness. To translate lip movements into words, first fill the system with input footage. Then use OpenCV to read frames and conduct face detection. Then use OpenCV to conduct lip detection. Following lip detection, the lip section of the frame will be cropped. In order to read the videos in the video dataset frame by frame once they have been loaded, the OpenCV library, an open-source toolkit for computer vision, machine learning, and image processing, is utilised. The next step is for an OpenCV-loaded haar cascade classifier to accept the frame and utilise it to recognise faces. The detected face image will be pass to shape predictor to get the coordinates for where the lips are located and get the lips cropped out.

The Hog method will divide the lip image i.e. cropped lip part in cells of size N × N pixels. The orientation of all pixels is computed and accumulated in an M-bins histogram of orientations. Finally, all cell histograms are concatenated in order to construct the final features vector. The SURF method (Speeded Up Robust Features) is a quick and reliable algorithm for local, similarity-invariant encoding and comparison of pictures. An area of interest (ROI) is first defined. The integral picture included by this ROI is then calculated. Lastly, the features are extracted from the integral image. The three features will be concatenated and append to an empty list in python, and also append the corresponding class label of that video to another empty list.

We will be using a GRU(Gated Recurrent Unit) based deep learning architecture, GRU networks are a subclass of RNN that effectively models sequential data by selectively updating the hidden state at each time step using gating methods. They have shown successful at a number of natural language processing tasks, including language modeling, machine translation, and speech recognition. After creating the model we need to compile the deep learning architecture with assigning loss functions and optimizers. With the concatenated features and their corresponding labels, we will perform train test split then we train our GRU model with train set and validate it with the test set . Finally, we will then compute the accuracy and store the model weights.

First, we will input a video for prediction. Next, the OpenCV library will read the video frame by frame. Finally, we will pass the frame to the Haar cascade classifier, which will detect faces. The detected face image will be pass to shape predictor and use it to determine the coordinates of the lips and clip out the portion of the lips. The lip's cropped portion will go through the Hog, SURF, and Haar feature extraction processes before being concatenated. The concatenated features are then input into our GRU model, which will then use them to predict a result.

V.METHODOLOGY

i. SURF

In computer vision, speeded up robust features (SURF) is a patented local feature detector and descriptor. It can be used for tasks such as object recognition, image registration, classification, or 3D reconstruction. It is partly inspired by the scale-invariant feature transform (SIFT) descriptor. The standard version of SURF is several times faster than SIFT and claimed by its authors to be more robust against different image transformations than SIFT.

To detect interest points, SURF uses an integer approximation of the determinant of Hessian blob detector, which can be computed with 3 integer operations using a precomputed integral image. Its feature descriptor is based on the sum of the Haar wavelet response around the point of interest. These can also be computed with the aid of the integral image.

SURF descriptors have been used to locate and recognize objects, people or faces, to reconstruct 3D scenes, to track objects and to extract points of interest.

The SURF algorithm is based on the same principles and steps as SIFT; but details in each step are different. The algorithm has three main parts: interest point detection, local neighbourhood description, and matching.

Detection:

SURF uses square-shaped filters as an approximation of Gaussian smoothing. (The SIFT approach uses cascaded filters to detect scale-invariant characteristic points, where the difference of Gaussians (DoG) is calculated on rescaled images progressively.) Filtering the image with a square is much faster if the integral image is used.

$$S(x,y) = \sum_{i=0}^{x} \sum_{j=0}^{y} I(i,j)$$

The sum of the original image within a rectangle can be evaluated quickly using the integral image, requiring evaluations at the rectangle's four corners.

SURF uses a blob detector based on the Hessian matrix to find points of interest. The determinant of the Hessian matrix is used as a measure of local change around the point and points are chosen where this determinant is maximal. In contrast to the Hessian-Laplacian detector by Mikolajczyk and Schmid, SURF also uses the determinant of the Hessian for selecting the scale, as is also done by Lindeberg. Given a point p=(x, y) in an image I, the Hessian matrix H(p, σ) at point p and scale σ, is:

$$H(p,\sigma) = \begin{pmatrix} L_{xx}(p,\sigma) & L_{xy}(p,\sigma) \\ L_{xy}(p,\sigma) & L_{yy}(p,\sigma) \end{pmatrix}$$

Scale-space representation and location of points of interest

Interest points can be found at different scales, partly because the search for correspondences often requires comparison images where they are seen at different scales. In other feature detection algorithms, the scale space is usually realized as an image pyramid. Images are repeatedly smoothed with a Gaussian filter, then they are subsampled to get the next higher level of the pyramid. Therefore, several floors or stairs with various measures of the masks are calculated:

$$\sigma_{\text{approx}} = \text{current filter size} \times \left(\frac{\text{base filter scale}}{\text{base filter size}} \right)$$

The scale space is divided into a number of octaves, where an octave refers to a series of response maps of covering a doubling of scale. In SURF, the lowest level of the scale space is obtained from the output of the 9×9 filters.

Hence, unlike previous methods, scale spaces in SURF are implemented by applying box filters of different sizes. Accordingly, the scale space is analyzed by up-scaling the filter size rather than iteratively reducing the image size. The output of the above 9×9 filter is considered as the initial scale layer at scale s =1.2 (corresponding to Gaussian derivatives with σ = 1.2). The following layers are obtained by filtering the image with gradually bigger masks, taking into account the discrete nature of integral images and the specific filter structure. This results in filters of size 9×9, 15×15, 21×21, 27×27,.... Non-maximum suppression in a 3×3×3 neighborhood is applied to localize interest points in the image and over scales. The maxima of the determinant of the Hessian matrix are then interpolated in scale and image space with the method proposed by Brown, et al. Scale space interpolation is especially important in this case, as the difference in scale between the first layers of every octave is relatively large.

Descriptor:

The goal of a descriptor is to provide a unique and robust description of an image feature, e.g., by describing the intensity distribution of the pixels within the neighbourhood of the point of interest. Most descriptors are thus computed in a local manner, hence a description is obtained for every point of interest identified previously.

The dimensionality of the descriptor has direct impact on both its computational complexity and point-matching robustness/accuracy. A short descriptor may be more robust against appearance variations, but may not offer sufficient discrimination and thus give too many false positives.

The first step consists of fixing a reproducible orientation based on information from a circular region around the interest point. Then we construct a square region aligned to the selected orientation, and extract the SURF descriptor from it.

Orientation Assignment:-

In order to achieve rotational invariance, the orientation of the point of interest needs to be found. The Haar wavelet responses in both x- and y-directions within a circular neighbourhood of radius 6s around the point of interest are computed, where s is the scale at which the point of interest was detected. The obtained responses are weighted by a Gaussian function centered at the point of interest, then plotted as points in a two-dimensional space, with the horizontal response in the abscissa and the vertical response in the ordinate. The dominant orientation is estimated by calculating the sum of all responses within a sliding orientation window of size π/3. The horizontal and vertical responses within the window are summed. The two summed responses then yield a local orientation vector. The longest such vector overall defines the orientation of the point of interest. The size of the sliding window is a parameter that has to be chosen carefully to achieve a desired balance between robustness and angular resolution.

Descriptor based on the sum of Haar wavelet responses:

To describe the region around the point, a square region is extracted, centered on the interest point and oriented along the orientation as selected above. The size of this window is 20s.

The interest region is split into smaller 4x4 square sub-regions, and for each one, the Haar wavelet responses are extracted at 5x5 regularly spaced sample points. The responses are weighted with a Gaussian (to offer more robustness for deformations, noise and translation).

Matching:

By comparing the descriptors obtained from different images, matching pairs can be found.

ii. Haar

A Haar-like feature is represented by taking a rectangular part of an image and dividing that rectangle into multiple parts. They are often visualized as black and white adjacent rectangles.Haar-like features are digital image features used in object recognition. They owe their name to their intuitive similarity with Haar wavelets and were used in the first real-time face detector.

Historically, working with only image intensities (i.e., the RGB pixel values at each and every pixel of image) made the task of feature calculation computationally expensive. A publication by Papageorgiou et el discussed working with an alternate feature set based on Haar wavelets instead of the usual image intensities. Paul Viola and Michael Jones adapted the idea of using Haar wavelets and developed the so-called Haar-like features. A Haar-like feature considers adjacent rectangular regions at a specific location in a detection window, sums up the pixel intensities in each region and calculates the difference between these sums. This difference is then used to categorize subsections of an image. For example, with a human face, it is a common observation that among all faces the region of the eyes is darker than the region of the cheeks. Therefore, a common Haar feature for face detection is a set of two adjacent rectangles that lie above the eye and the cheek region. The position of these rectangles is defined relative to a detection window that acts like a bounding box to the target object (the face in this case).

In the detection phase of the Viola–Jones object detection framework, a window of the target size is moved over the input image, and for each subsection of the image the Haar-like feature is calculated. This difference is then compared to a learned threshold that separates non-objects from objects. Because such a Haar-like feature is only a weak learner or classifier (its detection quality is slightly better than random guessing) a large number of Haar-like features are necessary to describe an object with sufficient accuracy. In the Viola–Jones object detection framework, the Haar-like features are therefore organized in something called a classifier cascade to form a strong learner or classifier.

The key advantage of a Haar-like feature over most other features is its calculation speed. Due to the use of integral images, a Haar-like feature of any size can be calculated in constant time (approximately 60 microprocessor instructions for a 2-rectangle feature).

VI. HOG

HOG, or Histogram of Oriented Gradients, is a feature descriptor that is often used to extract features from image data. It is widely used in computer vision tasks for object detection.

The histogram of oriented gradients (HOG) is a feature descriptor used in computer vision and image processing for the purpose of object detection. The technique counts occurrences of gradient orientation in localized portions of an image. This method is similar to that of edge orientation histograms, scale-invariant feature transform descriptors, and shape contexts, but differs in that it is computed on a dense grid of uniformly spaced cells and uses overlapping local contrast normalization for improved accuracy.

Robert K. McConnell of Wayland Research Inc. first described the concepts behind HOG without using the term HOG in a patent application in 1986. In 1994 the concepts were used by Mitsubishi Electric Research Laboratories. However, usage only became widespread in 2005 when Navneet Dalal and Bill Triggs, researchers for the French National Institute for Research in Computer Science and Automation (INRIA), presented their supplementary work on HOG descriptors at the Conference on Computer Vision and Pattern Recognition (CVPR). In this work they focused on pedestrian detection in static images, although since then they expanded their tests to include human detection in videos, as well as to a variety of common animals and vehicles in static imagery.

The essential thought behind the histogram of oriented gradients descriptor is that local object appearance and shape within an image can be described by the distribution of intensity gradients or edge directions. The image is divided into small connected regions called cells, and for the pixels within each cell, a histogram of gradient directions is compiled. The descriptor is the concatenation of these histograms. For improved accuracy, the local histograms can be contrast-normalized by calculating a measure of the intensity across a larger region of the image, called a block, and then using this value to normalize all cells within the block. This normalization results in better invariance to changes in illumination and shadowing.

The HOG descriptor has a few key advantages over other descriptors. Since it operates on local cells, it is invariant to geometric and photometric transformations, except for object orientation. Such changes would only appear in larger spatial regions. Moreover, as Dalal and Triggs discovered, coarse spatial sampling, fine orientation sampling, and strong local photometric normalization permits the individual body movement of pedestrians to be ignored so long as they maintain a roughly upright position. The HOG descriptor is thus particularly suited for human detection in images.

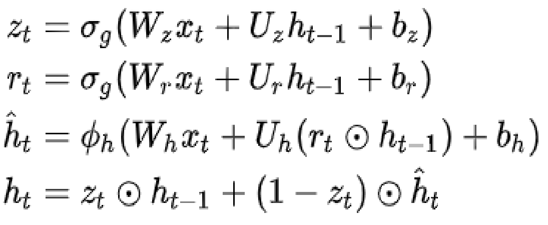
iii.Gated Recurrent Unit

Gated recurrent units(GRUs) are a gating mechanism in recurrent neural networks, introduced in 2014 by Kyunghyun Cho et al. The GRU is like a long short-term memory (LSTM) with a forget gate, but has fewer parameters than LSTM, as it lacks an output gate. GRU's performance on certain tasks of polyphonic music modeling, speech signal modeling and natural language processing was found to be similar to that of LSTM. GRUs shown that gating is indeed helpful in general and Bengio's team concluding that no concrete conclusion on which of the two gating units was better.

There are several variations on the full gated unit, with gating done using the previous hidden state and the bias in various combinations, and a simplified form called minimal gated unit.

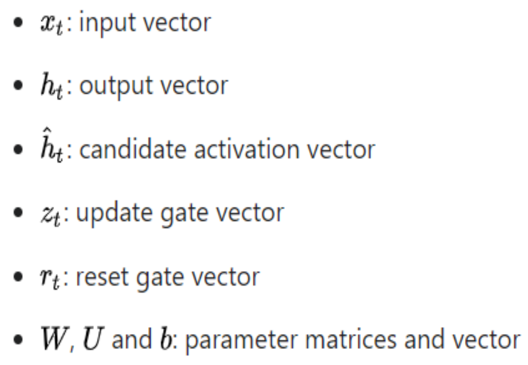
Fully gated unit:

Initially for t=0,the output vector h0=0



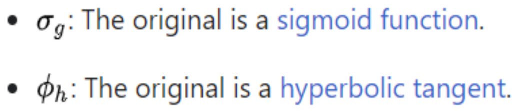
$$z_t = \sigma_g(W_z x_t + U_z h_{t-1} + b_z)$$
$$r_t = \sigma_g(W_r x_t + U_r h_{t-1} + b_r)$$
$$\hat{h}_t = \phi_h(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h)$$
$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \hat{h}_t$$

variables.

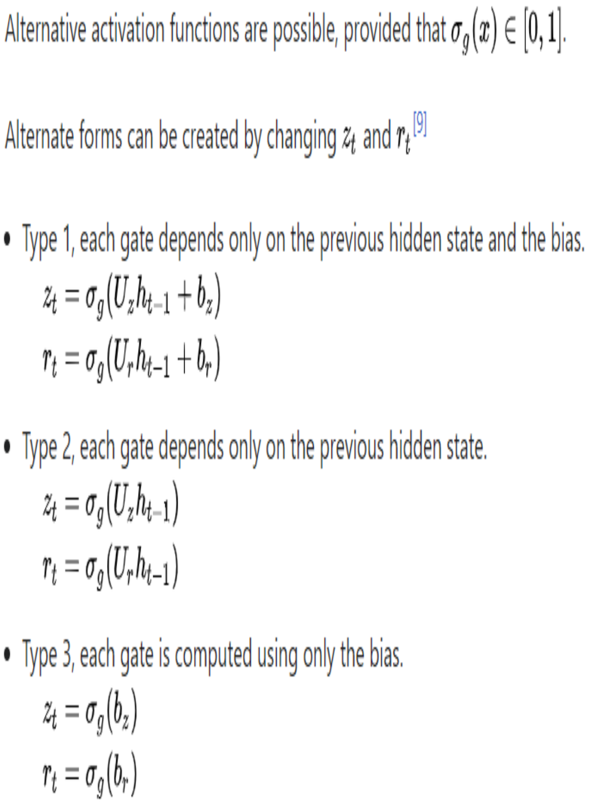


- x_t: input vector
- h_t: output vector
- $\hat{h}_t$: candidate activation vector
- z_t: update gate vector
- r_t: reset gate vector
- W, U and b: parameter matrices and vector

Activation functions



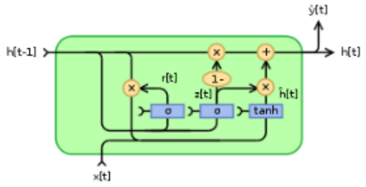
- σ_g: The original is a sigmoid function.
- ϕ_h: The original is a hyperbolic tangent.



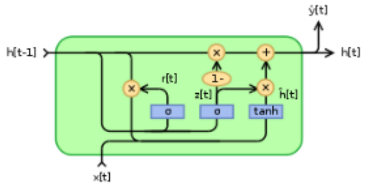
Alternative activation functions are possible, provided that $\sigma_g(x) \in [0,1]$.

Alternate forms can be created by changing z_t and r_t [9]

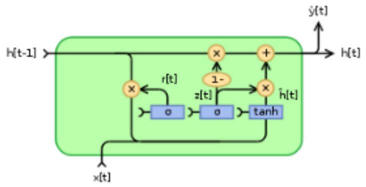
- Type 1, each gate depends only on the previous hidden state and the bias.
$$z_t = \sigma_g(U_z h_{t-1} + b_z)$$
$$r_t = \sigma_g(U_r h_{t-1} + b_r)$$
- Type 2, each gate depends only on the previous hidden state.
$$z_t = \sigma_g(U_z h_{t-1})$$
$$r_t = \sigma_g(U_r h_{t-1})$$
- Type 3, each gate is computed using only the bias.
$$z_t = \sigma_g(b_z)$$
$$r_t = \sigma_g(b_r)$$



Type 1



Type 2



Type 3

Minimal gated unit

The minimal gated unit is similar to the fully gated unit, except the update and reset gate vector is merged into a forget gate. This also implies that the equation for the output vector must be changed:

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f)$$
$$\hat{h}_t = \phi_h(W_h x_t + U_h(f_t \odot h_{t-1}) + b_h)$$
$$h_t = (1 - f_t) \odot h_{t-1} + f_t \odot \hat{h}_t$$

Variables

- x_t: input vector
- h_t: output vector
- $\hat{h}_t$: candidate activation vector
- f_t: forget vector
- W, U and b: parameter matrices and vector

VI. RESULT AND DISCUSSION

Here we have training and testing accuracy and loss in terms of number of epochs. The losses decreases and accuracy increases with each epochs.

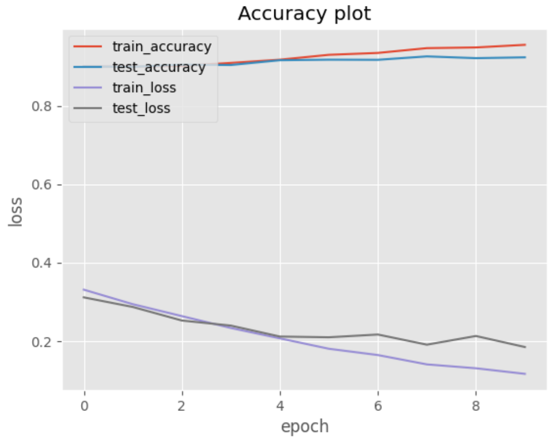


Figure 3: Accuracy plot graph

Training	96-97%
Test	90-91%

Table.1: Model Accuracy

Here, we use the MIRACL- VC1 dataset, which includes both training and testing data. For training set accuracy, we got 96–97%, and for test set accuracy, 90–91%. By employing a productive system and training the model on a substantial amount of data, we can enhance the performance of our model. As a result, we can develop an effective model that operates in a real-time environment.

VII. CONCLUSION

In this paper we proposed a deep learning based model to recognize the lip movement and to convert it into text format. We employed SURF,Hog,Haar algorithms to this problem. Here, we designed a desktop application that is very user-friendly and extremely helpful for those with hearing impairments. Enhancement in lip reading increases the possibility to allow better speech recognition in noisy or loud environment. A prominent benefit could be the development for people with hearing disabilities. Similarly for security purposes, a lip reading system can be applied for speech analysis to determine and predict information from the speaker when audio is corrupted or absent in the video. In this system, the input video is first converted into frames at this level. On those frames, perform a face detection operation. After face detection, use a shape predictor to trim the lip area from the image frames. The three algorithms Hog, SURF, and Haar extract features. The features from each feature extraction algorithm are then combined. Pass the data to the trained model after loading it, and it will predict the term. The model finally generates an output as English words.

Acknowledgment

The outcome of this paper required a lot of guidance and support. We are so privileged to have this support all along the completion of this paper. All that we have done is only with the supervision and guidance given and we are thankful for that. We respect and thank Mrs.Gishma K.M and Universal Engineering College, for giving all the support and guidance for this paper. We are thankful and fortunate enough to get thisencouragement, support and guidance.

References

[1] A. Fernandez-Lopez and M. F. Sukno, "Survey on automatic lip-reading in the era of deep learning", Image Vis. Comput., vol. 78, pp. 53-72, Oct. 2018.

[2] D. Bhatt, C. Patel, H. Talsania, J. Patel, R. Vaghela, S. Pandya, et al., "CNN variants for computer vision: History architecture application challenges and future scope", Electronics, vol. 10, no. 20, pp. 2470, Oct. 2021.

[3] C. Patel, D. Bhatt, U. Sharma, R. Patel, S. Pandya, K. Modi, et al., "DBGC: Dimension-based generic convolution block for object recognition", Sensors, vol. 22, no. 5, pp. 1780, Feb. 2022.

[4] T. Stafylakis and G. Tzimiropoulos, ‘‘Combining residual networks with LSTMs for lipreading,’’ in Proc. Interspeech, Aug. 2017, pp. 3652–3656.

[5] K. Hara, H. Kataoka and Y. Satoh, "Learning spatio-temporal features with 3D residual networks for action recognition", Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW), pp. 3154-3160, Oct. 2017.

[6] J. Xiao, S. Yang, Y. Zhang, S. Shan and X. Chen, "Deformation flow based two-stream network for lip reading", Proc. 15th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), pp. 364-370, Nov. 2020..

[7] Y. M. Assael, B. Shillingford, S. Whiteson, and N. de Freitas, ‘‘LipNet: End-to-end sentence level lipreading,’’ in Proc. ICLR Conf., 2016,pp. 1–13..

[8] B. Shillingford, Y. Assael, M. Hoffman, T. Paine, C. Hughes, U. PrabhuH. Liao, H. Sak, R. Hasim, H. Rao, L. Bennett, M. Mulville, B. Coppin,B. Laurie, A. Senior, and N. Freitas, ‘‘Large-scale visual speech recognition,’’ in Proc. INTERSPEECH, 2018..

[9] K. Thangthai, H. L. Bear, and R. Harvey, ‘‘Comparing phonemes and visemes with DNN-based lipreading,’’ in Proc. Brit. Mach. Vis. Conf., 2017, pp. 1–11.

[10] A. B. Mattos, D. Oliveira, and E. Morais, ‘‘Improving viseme recognition using GAN-based frontal view mapping,’’ in Proc. Analysis and Modeling of Faces and Gestures (CVPR), Jun. 2018, pp. 2148–2155.

[11] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, et al., " An image is worth 16times16 words: Transformers for image recognition at scale ", Proc. Int. Conf. Learn. Represent., 2021.

[12] S. Fenghour, D. Chen, and P. Xiao, ‘‘Decoder-encoder LSTM for lip read-ing,’’ in Proc. 8th Int. Conf. Softw. Inf. Eng. - ICSIE, 2019, pp. 162–166.

[13] H. Wang, G. Pu and T. Chen, "A Lip Reading Method Based on 3D Convolutional Visio-n Transformer," in IEEE Access, vol. 10, pp. 77205-77212, 2022, doi: 10.1109/ACCESS.2022.3193231

[14] M. Luo, S. Yang, S. Shan and X. Chen, "Synchronous bidirectional learning for multilingual lip reading", Proc. BMVC, 2020

[15] J. S. Son and A. Zisserman, ‘‘Lip reading in profile,’’ in Proc. Brit. Mach.Vis. Conf., 2017, pp. 1–11.

[16] H. Wang, G. Pu and T. Chen, "A Lip Reading Method Based on 3D Convolutional Vision Transformer," in IEEE Access, vol. 10, pp. 77205-77212, 2022, doi: 10.1109/ACCESS.2022.3193231.

[17] J. S. Son and A. Zisserman, ‘‘Lip reading in profile,’’ in Proc. Brit. Mach.Vis. Conf., 2017, pp. 1–11.

[18] S. Petridis, Y. Wang, Z. Li, and M. Pantic, ‘‘End-to-end multi-view lipreading,’’ in Proc. Brit. Mach. Vis. Conf., 2017, pp. 1–17.

[19] C. Wang, ‘‘Multi-grained spatio-temporal modelling for lip-reading,’’ in Proc. Brit. Mach. Vis. Conf., 2019, pp. 1–11.

[20] M. A. Abrar, A. N. M. N. Islam, M. M. Hassan, M. T. Islam, C. Shahnaz and S. A. Fattah, "Deep Lip Reading-A Deep Learning Based Lip-Reading Software for the Hearing Impaired," 2019 IEEE R10 Humanitarian Technology Conference (R10-HTC)(47129), 2019, pp. 40-44, doi: 10.1109/R10-HTC47129.2019.9042439.

[21] X. Weng and K. Kitani, ‘‘Learning spatio-temporal features with two-stream deep 3D CNNs for lip reading,’’ in Proc. Brit. Mach. Vis. Conf.,2019, pp. 1–13.

[22] X. Zhang, F. Cheng and W. Shilin, "Spatio-temporal fusion based convolutional sequence learning for lip reading", Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), pp. 713-722, Oct. 2019.

[23] H. Wu, B. Xiao, N. Codella, M. Liu, X. Dai, L. Yuan, et al., "CvT: Introducing convolutions to vision transformers", arXiv:2103.15808, 2021.

[24] I. Fung and B. Mak, ‘‘End-to-end low-resource lip-reading with maxout CNN and LSTM,’’ in Proc. IEEE Int. Conf. Acoust., Speech Signal Process.(ICASSP), Apr. 2018, pp. 2511–2515.

[25] S. Petridis, Y. Wang, Z. Li, and M. Pantic, ‘‘End-to-end audiovisual fusion with LSTMs,’’ in Proc. 14th Int. Conf. Auditory-Visual Speech Process.,Aug. 2017, pp. 1–5.

[26] S. Fenghour, D. Chen, K. Guo and P. Xiao, "Lip Reading Sentences Using Deep Learning With Only Visual Cues," in IEEE Access, vol. 8, pp. 215516-215530, 2020, doi: 10.1109/ACCESS.2020.3040906.