Documentation paper / Project report

Visual Speech Recognition Model

Adviser	Bálint Gyires Tóth	
Author	Ágnes Márkó	Dávid Kocsis
E-Mail	marko.agi13@gmail.com	kocsisd05@gmail.com
Course of study	Aquincum Institute of Technology - Deep Learning	
Date of submission	05/15/2024	

Table of contents

Abstract

Lip reading is a technique to understand words or speech by visual interpretation of face, mouth, and lip movement without the involvement of audio. This project verifies the use of machine learning by applying deep learning and neural networks to devise an automated lip-reading system. A subset of the dataset was trained on four separate architectures. The trained lip reading models were evaluated based on their accuracy to predict words and saved after the training.

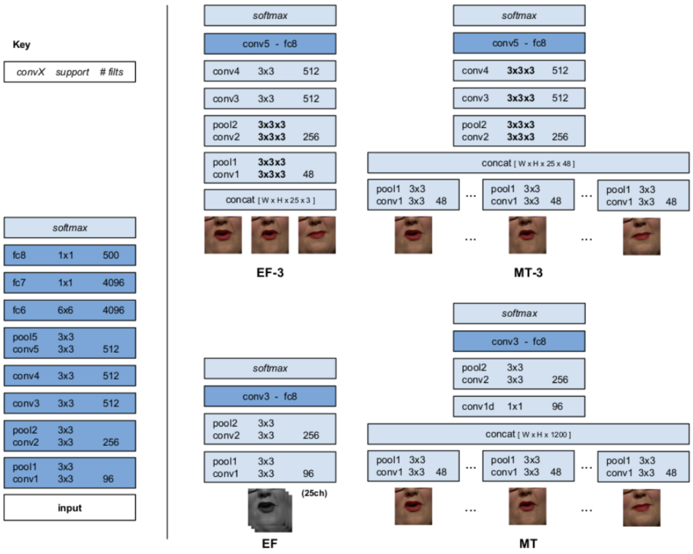
1. Introduction

Currently more than 1.5 billion people live with hearing loss, which is nearly 20% of the global population and it is expected that by 2050, there could be over 2.5 billion people to have some degree of hearing loss. That is a reason why lip reading has received increasing attention in recent years. Visual Speech Recognition (VSR), also known as Automatic Lip-Reading (ALR) is a process, which aims to recognize the content of speech based on lip movements. In recent years many deep learning based methods have been proposed, besides traditional machine learning to work on the problem of ALR.

In this project we would like to present an approach for determining spoken words from a visual dataset of known words. As for now we would like to build a model that can recognize a fixed number of words presented as sequences of lip movements, which can be improved over time.

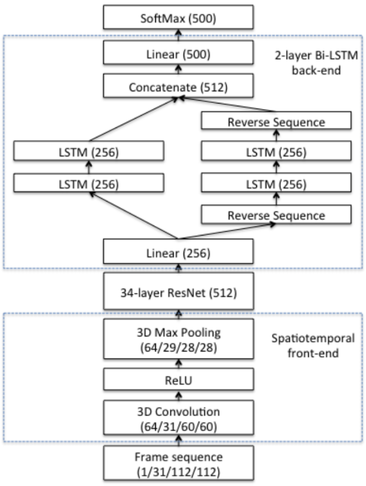
2. Previous solutions

Lip Reading in the Wild [1], authored by a team from the University of Oxford, proposes different CNN architectures that are able to recognize words from their dataset. They test the performance of 4 different architectures, two 2D-CNNs and two 3D-CNNs, and achieve a top recognition rate of 61.1% using a 2D CNN architecture with multiple input towers that are concatenated after performing one convolution on each frame. The architectures used:

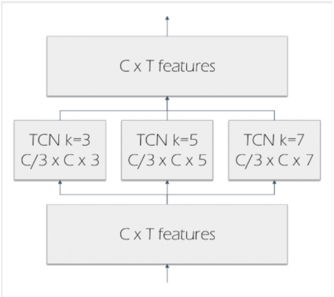


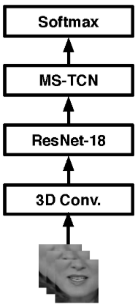
The authors consider their architectures 2D and 3D even though the inputs to each are 3D and 4D respectively. The EF-3 and MT-3 architectures take a collection of color images thus having the dimension of WxHx25x3 where W is the width of each frame, H is the height of each frame, 25 is the number of total frames per video (the temporal dimension), and 3 is the multi-channels of the color image.

In 2017 Stafylakis and Tzimiropoulos propose an end-to-end deep learning architecture for word-level visual speech recognition. [2] The system is a combination of spatiotemporal convolutional, residual and bidirectional Long Short-Term Memory networks. They train and evaluate it on the Lipreading In-The-Wild dataset. The proposed network achieves word accuracy equal to 83.0%, yielding 6.8% absolute improvement over the current state-of-the-art, without using information about word boundaries during training or testing. The block-diagram of the proposed network:

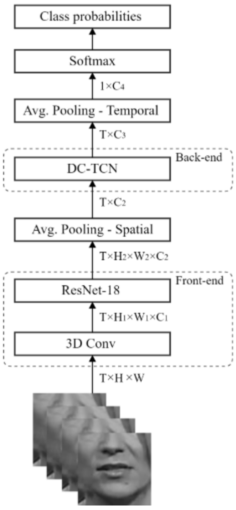


In 2020 Martinez, Ma, Petridis and Pantic constructed a word-based lip-reading system[3] with a frontend that entails a spatiotemporal CNN followed by a ResNet-18 CNN. For the backend, they proposed called a Multi-Scale Temporal Convolutional Network (MS-TCN), devised to tailor the receptive field of a TCN so that long and short term information can be mixed up. A MS-TCN block consists of a series of TCNs, each with a different kernel size whereby the outputs are concatenated. Their system was trained and evaluated on the English datasets LRW and Mandarin dataset LRW-1000 achieving word accuracies of 85.3% and 41.4% respectively. The Multi-Scale CNN and the overall architecture:





In the same year a different team proposed modifications to the system of Martinez et al. by using a Densely Connected Temporal Convolutional Network (DC-TCN) [4] instead of the MS-TCN [3]contained within the frontend for the aim of providing denser and more robust temporal features. Their approach utilizes the Squeeze-and-Excitation block, a light-weight attention mechanism, to further enhance the model’s classification power. The DC-TCN method has achieved 88.36 % accuracy on the Lip Reading in the Wild (LRW) dataset and 43.65 % on the LRW-1000 dataset. The general framework of their method:



3. Dataset

In this project we used the Oxford-BBC Lip Reading in the Wild (LRW) dataset [1] for training and evaluation of the deep learning models. LRW dataset is a large publicly available (upon request) dataset for non-commercial and academic researches. The dataset consists of short video clip segments of 1.16 seconds (approximately 29 frames). It comprises of hundreds of speakers videos from BBC programs, primarily from talk shows and news. Each video segment is clipped to record only the part where the speaker utters the word. Metadata is provided in the dataset to determine the start and end frames for each word duration in the video. This dataset contains 500 different word instances with up to 1000 utterances for each word spoken by different speakers and was compiled into a training set, validation set and test set.

Due to the limited computational resource we only trained a part of the LRW dataset. Training the whole dataset within our computer architecture would take several days. We selected 6 words from the LRW dataset to train the lip reading model. The selected words have a noticeable difference in articulation and lip movement. The selected words from the LRW dataset are:

4. Proposed Method

4.1. Data pre-processing

In the pre-processing stage, the speaker videos from LRW was initially used to detect the mouth - Region of Interest (ROI). An OpenCV python framework with Haar Feature-Based Cascade classifier was used to detect face region from each input videos. OpenCV is an open-source multi-platform library of programming functions focused on computer vision.

The lip detector function is built primarily off the dlib package. It will also make use of the OpenCV library for handling video files, and also imutils, an open source library, which is a series of OpenCV convenience functions. For this project we are using the 68 point facial landmark detector. The previously detected face objects are the input into the shape predictor to identify the locations of the landmark. The output of the shape predictor is converted to a Numpy array for convenience, using the imutils library. This shape array that contains locations of the facial landmarks is then used to locate the lips in the image. The landmarks that correspond to the lips are 48-68, only that subset of the shape array is used. That subset is then converted to a numpy array and used as an input in OpenCV's bounding rectangle function. A margin of 10 pixels is included to grab excess information around the lips. This bounding rectangle is then applied to the frame to grab the required lip data. Lastly, the lip data is resized.

4.2. Model Architectures

The first model is a 3D convolutional neural network (CNN) with residual connections and WaveNet activation. The model consists of four blocks, each containing multiple 3D convolutional layers, batch normalization, and rectified linear unit (ReLU) activation functions. The residual connections help alleviate the vanishing gradient problem and improve the flow of information through the network.

The first block consists of a 3D convolutional layer with 64 filters, followed by a ReLU activation and batch normalization. The output is then passed through a max-pooling layer. The second and third blocks each contain two 3D convolutional layers with varying filter sizes and strides, followed by ReLU activations and batch normalization. The output of these blocks is added to the input of the block, forming the residual connection. The fourth block introduces a novel approach by incorporating the WaveNet activation function, which is known for its ability to model long-term dependencies in sequential data. The block consists of three 3D convolutional layers with varying dilation rates, followed by WaveNet activations and spatial dropout. The output of these layers is added to the input, forming the residual connection. The features are then flattened and passed through a dense layer before being fed into the output layer with a softmax activation function.

The second model is a 3D convolutional neural network with ResNet blocks and bidirectional LSTM (BiLSTM) units. The model consists of four blocks, with the first block containing a 3D convolutional layer, ReLU activation, and batch normalization. The second and third blocks each contain two 3D convolutional layers with varying filter sizes and strides, followed by ReLU activations and batch normalization. The output of these blocks is added to the input of the block, forming the residual connection.The output is then reshaped and fed into a BiLSTM layer. The output of the BiLSTM layer is passed through a dropout layer, another BiLSTM layer, and finally a dense layer before being fed into the output layer with a softmax activation function.

For the third model we propose a deep learning model that combines convolutional neural networks (CNNs) and temporal convolutional networks (TCNs) with squeeze-and-excitation (SE) attention mechanisms for sequence learning. The model is designed to learn spatial-temporal features from a sequence of raw visual speech frames, and subsequently classify them into phonetic categories.

The proposed model consists of five convolutional blocks, followed by a series of TCN blocks and SE attention layers. Each convolutional block includes a 3D convolutional layer with batch normalization and ReLU activation, followed by a 3D convolutional layer with a smaller kernel size and no strides, and another batch normalization layer. This design allows the model to learn hierarchical spatial-temporal features from the input visual speech frames. After the convolutional blocks, we use a reshaping layer to flatten the spatial dimensions of the output feature map and obtain a sequence of feature vectors. We then apply a series of TCN blocks with increasing dilation rates to capture long-range temporal dependencies in the feature sequence. Each TCN block includes a 1D convolutional layer with causal padding, batch normalization, and ReLU activation, followed by another 1D convolutional layer with the same configuration and a dropout layer. We also include a downsampling layer in the first TCN block to reduce the sequence length.

To further enhance the model's ability to learn sequence representations, we incorporate SE attention layers after each TCN block. The SE attention mechanism uses a gating mechanism to weight the importance of each feature vector in the sequence, allowing the model to focus on the most relevant features for the phonetic classification task.

In summary, the proposed model leverages the strengths of CNNs and TCNs to learn spatial-temporal features from visual speech frames and capture long-range temporal dependencies in the feature sequence. The SE attention mechanism further enhances the model's ability to learn sequence representations, resulting in improved phonetic classification accuracy.

5. Evaluation Method

We utilized two commonly used metrics in classification tasks: classification report and confusion matrix. These metrics provide a comprehensive overview of the model's performance, including precision, recall, F1-score, and support for each phonetic category, as well as the overall accuracy.

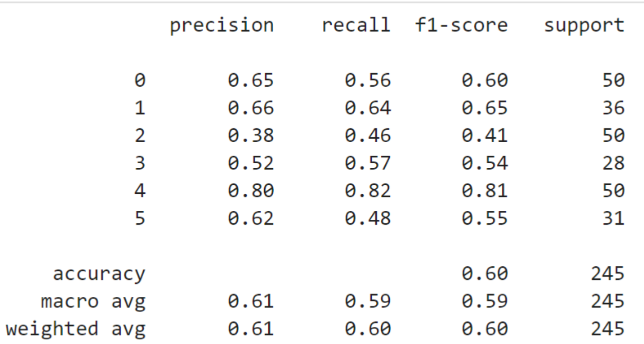
First, we used the scikit-learn library to obtain a summary of the model's performance on the test set. This calculated the precision, recall, F1-score, and support for each phonetic category, as well as the weighted average of these metrics across all categories. The precision measures the proportion of true positive predictions among all positive predictions, while the recall measures the proportion of true positive predictions among all actual positive instances. The F1-score is the harmonic mean of precision and recall, and the support measures the number of instances in each phonetic category.

Next, we used confusion matrix of the model's predictions on the test set. The confusion matrix is a table that summarizes the number of true positive, false positive, false negative, and true negative predictions for each phonetic category. We visualize the confusion matrix using the seaborn library, which provides a heatmap representation that highlights the areas of confusion and the overall accuracy.

These evaluation metrics provide valuable insights into the model's performance and help us identify areas for improvement. By examining the classification report and confusion matrix, we can determine which phonetic categories are more challenging for the model to recognize and adjust the model architecture and training parameters accordingly.

6. Results

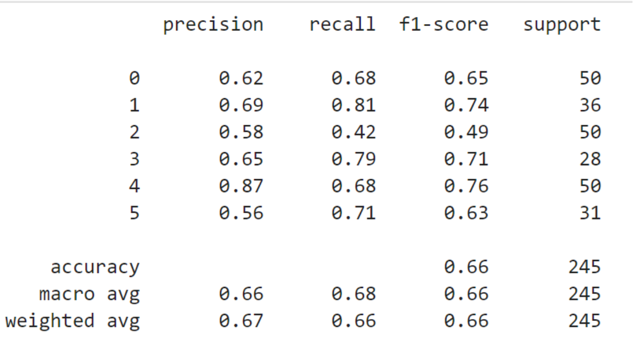
The results of each model are presented in the tables below. For each word in the simpler dataset we used for training and testing, it was calculated the precision, recall, f1-score and support. The summary section at the end of the report includes the overall accuracies of the models.



```
              precision    recall  f1-score   support

           0       0.65      0.56      0.60        50
           1       0.66      0.64      0.65        36
           2       0.38      0.46      0.41        50
           3       0.52      0.57      0.54        28
           4       0.80      0.82      0.81        50
           5       0.62      0.48      0.55        31

    accuracy                           0.60       245
   macro avg       0.61      0.59      0.59       245
weighted avg       0.61      0.60      0.60       245
```

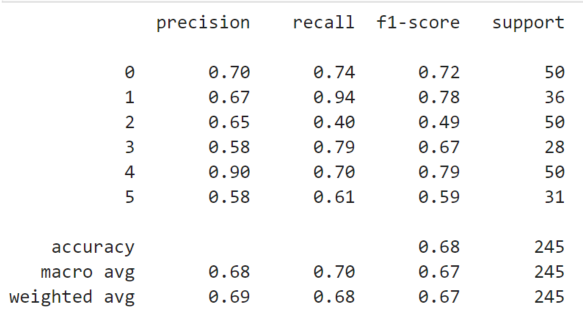


```
              precision    recall  f1-score   support

           0       0.62      0.68      0.65        50
           1       0.69      0.81      0.74        36
           2       0.58      0.42      0.49        50
           3       0.65      0.79      0.71        28
           4       0.87      0.68      0.76        50
           5       0.56      0.71      0.63        31

    accuracy                           0.66       245
   macro avg       0.66      0.68      0.66       245
weighted avg       0.67      0.66      0.66       245
```

First model: Dilated Convnet *Second model:BiLSTM model*



```
              precision    recall  f1-score   support

           0       0.70      0.74      0.72        50
           1       0.67      0.94      0.78        36
           2       0.65      0.40      0.49        50
           3       0.58      0.79      0.67        28
           4       0.90      0.70      0.79        50
           5       0.58      0.61      0.59        31

    accuracy                           0.68       245
   macro avg       0.68      0.70      0.67       245
weighted avg       0.69      0.68      0.67       245
```

Third model: Temporal Convnet

7. Discussion

Several conclusions can be drawn from the results presented above. First, the Dilated Convnet model (first model) yields a word accuracy of 60.0%, demonstrating a baseline performance for visual speech recognition tasks.

In comparison, the BiLSTM model (second model) achieves a 6.0% absolute improvement over the Dilated Convnet model, highlighting the effectiveness of recurrent neural networks in modeling temporal sequences of visual speech data. The BiLSTM model's ability to capture long-term dependencies in the data leads to improved performance in recognizing spoken words.

Furthermore, the Temporal Convnet model (third model) outperforms both the Dilated Convnet and BiLSTM models, achieving a word accuracy of 68.0%. This improvement can be attributed to the Temporal Convnet's ability to model short-term dynamics in the mouth region, demonstrating the importance of incorporating temporal information in visual speech recognition tasks.

The results also suggest that the choice of model architecture has a significant impact on performance. The BiLSTM model, with its ability to capture long-term dependencies, outperforms the Dilated Convnet model, which relies on convolutional layers to extract features. Similarly, the Temporal Convnet model, which incorporates temporal information, outperforms both the Dilated Convnet and BiLSTM models.

Overall, the results demonstrate the effectiveness of deep learning models in visual speech recognition tasks and highlight the importance of incorporating temporal information and modeling long-term dependencies in the data. The findings of this study can inform the development of more accurate and robust visual speech recognition systems.

References

[1] Joon Son Chung and Andrew Zissermanm, 2016 ,Lip Reading in the Wild, Visual Geometry Group, Department of Engineering Science, University of Oxford

[2] T. Stafylakis and G. Tzimiropoulos, "Combining residual networks with LSTMs for lipreading", Proc. Interspeech, pp. 3652-3656, Aug. 2017

[3] B. Martinez, P. Ma, S. Petridis and M. Pantic, "Lipreading using temporal convolutional networks", Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP), pp. 6319-6323, May 2020.

[4]Pingchuan Ma, Yujiang Wang, Jie Shen, Stavros Petridis and Maja Pantic, Lip-reading with Densely Connected Temporal Convolutional Networks, 2021 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 2856-2865, 2021