Real Time Visual Speech Recognition

**Group Name:** CrypTAUX

**Group Members:**

Mitali Kambli (J025)

J. Sharat Shankar (J045)

Gayathri Shrikanth (J046)

# Index

|  |  |
| --- | --- |
| Sr no. | Title |
| 1. | [INTRODUCTION](#_1nt5f3yycvom) |
| 2. | [RELATED WORK](#_i6j0bzqn001y) |
| 3. | [PROBLEM STATEMENT](#_r53ojci23j5w) |
| 4. | [OBJECTIVES](#_apeuwlku7xtv) |
| 5. | [METHODOLOGY (DIAGRAM)](#_lmlojfkkdg8) |
| 6. | [ALGORITHM FLOWCHART](#_6cjhqudfyvce) |
| 7. | [IMPLEMENTATION AND RESULT](#_jtjawelykbly) |
| 8. | [CONCLUSION AND RECOMMENDATION](#_aglpbp14ia5y) |
| 9. | [REFERENCES](#_prdy5uvvrei4) |

# **INTRODUCTION**

Machine learning methods have had a great impact on social progress in recent years, which promoted the rapid development of artiﬁcial intelligence technology and solved many practical problems [1].

Lip-reading technology is one of the important components of human–computer interaction technology, virtual reality (VR) technology, information security, speech recognition and assisted driving systems. The research in lip-reading involves many ﬁelds, such as pattern recognition, computer vision, natural language comprehension and image processing Lip-reading is typically, visually interpreting the movements of a speaker’s lip during speaking with no accompanying audio data.

Humans generally display a poor ability to lip read, hearing-impaired people achieve an accuracy of only 17±12% even for a limited subset of 30 monosyllabic words and 21±11% for 30 compound words (Easton & Basala, 1982). An important goal, therefore, is to automate lipreading. Machine lip-readers have enormous practical potential, with applications in improved hearing aids, silent dictation in public spaces, security, speech recognition in noisy environments, biometric identiﬁcation, and silent-movie processing and the captioning of silent films and videos [2].

It would thus be beneficial to automate the task of lip reading. From a mathematical point of view, it requires converting the mouth movements to a truthful representation for possible visual recognition. Experiments over many years have revealed that speech intelligibility increases if both the audio and visual information are available [3].

It plays a vital role in human language communication and visual perception. Especially in noisy environments or VR environments, visual signals can remove redundant information, complement speech information, increase the multi-modal input dimension of immersive interaction, reduce the time and workload of learning lip language and lip movement on humans, and improve automatic speech recognition ability.

# 

# **RELATED WORK**

Automating lip-reading is a complicated process, including the coarticulation effect, visual features diversity and speaker-dependency of features. The history of these issues along with some proposed solutions for them can be found in the literature. [5]

**Highlighting speaker-dependency:**

In (Ngiam et al., 2011), the PCA is applied to the mouth region-of-interest (ROI), and a deep autoencoder (AE) is trained to extract the bottleneck (BN) features. The features from the entire utterance were fed to a support vector machine ignoring the temporal dynamics of the speech. Digit classification accuracies of 64.4% and 68.7% are achieved in this study, using AVLetters and CUAVE databases, respectively.

In (Srivastava and Salakhutdinov, 2012), similar feature extraction approach proposed by (Ngiam et al., 2011) is followed; this is done, while a deep Boltzmann Machine (DBM) is trained to extract multimodal representations. The employed features are extracted from the entire utterance and fed to a support vector machine. Working on CUAVE database, digit classification accuracy of 69% is achieved.

In (Noda et al., 2014), a convolutional neural network (CNN) is proposed to act as the feature extraction block for a lip-reading system; the speaker’s mouth area images along with the phoneme labels are used during the training phase. The evaluation is done on an audio-visual speech database comprising 300 Japanese words with six different speakers, each of which is modeled with an independent CNN.

In (Mroueh et al., 2015), some deep multimodal learning methods are proposed to fuse speech and visual modalities in an AVSR system. Two unimodal deep networks are first trained separately over the audio and video data, and their final hidden layers are fused to form a joint feature which is further used in another deep network. The experiments are conducted on the IBM large vocabulary audio-visual studio database, leading to phone error rate (PER) of 69.36% and 41% in video-only and audio-only DNN, respectively.

In (Almajai et al., 2016), a SI lip-reading system is designed, using Resource Management (RM) database. A combination of MLLT followed by SAT is applied over the CSAM features introduced in (Lan et al., 2012b). HMM models are trained over phoneme and viseme units, but the phoneme implementation represents the superiority. The best word accuracies

of 45% and 54% are achieved with HMM and DNN application, respectively.

**Automated lip reading:**

Most existing work on lipreading does not employ deep learning. Such work requires either heavy preprocessing of frames to extract image features, temporal preprocessing of frames to extract video features, or other types of handcrafted vision pipelines. The automated lip reading literature is too vast to adequately cover, so we refer the reader to Zhou et al. (2014) for an extensive review.

Goldschenetal(1997) were the ﬁrst to do visual-only sentence-level lipreading using hidden Markov models (HMMs) in a limited dataset, using hand-segmented phones. Later, Neti et al. (2000) were the ﬁrst to do sentence-level audiovisual speech recognition using an HMM combined with hand-engineered features, on the IBM ViaVoice (Neti et al., 2000) dataset. The authors improve speech recognition performance in noisy environments by fusing visual features with audio ones. The dataset contains 17111 utterances of 261 speakers for training (about 34.9 hours) and is not publicly available. As stated, their visual-only results cannot be interpreted as visual-only recognition, as they are used as rescoring of the noisy audio-only lattices. Using a similar approach, Potamianos et al. (2003) report speaker independent and speaker adapted 38.53%, 16.77% WER in the connected DIGIT corpus, which contains sentences of digits.

Furthermore, Gergen et al. (2016) use speaker-dependent training on an LDA-transformed version of the Discrete Cosine Transforms of the mouth regions in an HMM/GMM system. This work holds the previous state-of-the-art on the GRID corpus with a speaker-dependent accuracy of 86.4%.

**Classiﬁcation with deep learning:**

In recent years, there have been several attempts to apply deep learning to lipreading. However, all of these approaches perform only word or phoneme classiﬁcation. Approaches include learning multimodal audio-visual representations (Ngiam et al., 2011; Sui et al., 2015; Ninomiya et al., 2015; Petridis & Pantic, 2016), learning visual features as part of a traditional speech-style processing pipeline for classifying words and/or phonemes (Almajai et al., 2016; Takashima et al., 2016; Noda et al., 2014; Koller et al., 2015), or combinations thereof (Takashima et al., 2016).

Chung & Zisserman (2016a) propose spatial and spatiotemporal convolutional neural networks, based on VGG, for word classiﬁcation. The architectures are evaluated on a word-level dataset BBC TV (333 and 500 classes), but, as reported, their spatiotemporal models fall short of the spatial architectures by an average of around 14%. Additionally, their models cannot handle variable sequence lengths and they do not attempt sentence-level sequence prediction. Chung & Zisserman (2016b) train an audio-visual max-margin matching model for learning pre-trained mouth features, which they use as inputs to an LSTM for 10-phrase classiﬁcation on the OuluVS2 dataset, as well as a non-lipreading task.

Wand et al. (2016) introduce LSTM recurrent neural networks for lipreading but address neither sentence-level sequence prediction nor speaker independence.

Garg et al. (2016) apply a VGG pre-trained on faces to classifying words and phrases from the MIRACL-VC1 dataset, which has only 10 words and 10 phrases. However, their best recurrent model is trained by freezing the VGG Net parameters and then training the RNN, rather than training them jointly. Their best model achieves only 56.0% word classiﬁcation accuracy, and 44.5% phrase classiﬁcation accuracy, despite both of these being 10-class classiﬁcation tasks.

**Sequence prediction in speech recognition:**

The ﬁeld of automatic speech recognition (ASR) would not be in the state it is today without modern advances in deep learning. The connectionist temporal classiﬁcation loss (CTC) of Graves et al. (2006) drove the movement from deep learning as a component of ASR, to deep ASR systems trained end-to-end (Graves & Jaitly, 2014;Maasetal.,2015;Amodeietal.,2015). As mentioned earlier, much recent lipreading progress has mirrored early progress in ASR, but stopping short of sequence prediction.

**Lipreading Datasets:**

Lipreading datasets (AVICar, AVLetters, AVLetters2, BBC TV, CUAVE, OuluVS1, OuluVS2) are plentiful (Zhou et al., 2014; Chung & Zisserman, 2016a). A popular dataset is the Lip Reading in the Wild Dataset which consists of up to 1000 utterances of 500 different words, spoken by hundreds of different speakers. All videos are 29 frames (1.16 seconds) in length, and the word occurs in the middle of the video. The only problem with such datasets is that they only contain single words or are too small. One exception is the GRID corpus (Cookeetal.,2006),which has audio and video recordings of 34 speakers who produced 1000 sentences each, for a total of 28 hours across 34000 sentences.

Table 1 summarises state-of-the-art performance in each of the main lipreading datasets. The size column represents the number of utterances used by the authors for training. Although the GRID corpus contains entire sentences, Gergen et al. (2016)C consider only the simpler case of predicting isolated words. [6]

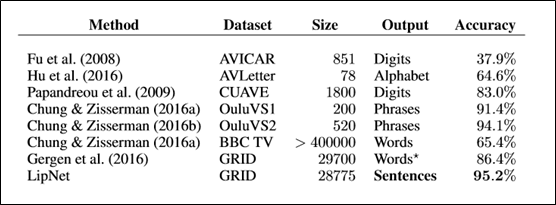


Table 1: Performance of Lip Reading methods on various datasets

# 

# **PROBLEM STATEMENT**

Classification problem of detecting what words are being spoken out of a fixed set of known words in order to achieve as high classification accuracy as possible in the test set.

# **OBJECTIVES**

Our objectives for automatic lip-reading recognition can be divided into five parts:

Firstly, we will extract keyframes from a sample video, use the key points of the mouth to locate the mouth area to reduce the complexity of redundant information and computational processing in successive frames and the data will be augmented in order to increase the size of the training dataset.

Then, features will be extracted from the original mouth image using the VGG16 network with the pre-trained weights of the ImageNet dataset. These extracted features will be made to pass through a small fully-connected network in order to train the dataset.

After the training is done, we will validate the model using the validation set which will follow the same pipeline for feature extraction as the training set and calculate the accuracy.

The final part is to implement a real time lip reader using the camera in our machines which will be able to capture a video stream of the speaker uttering the word and predict the word uttered by a user using the trained lip reading model.

# 

# **METHODOLOGY**

# **Data Pre-processing:** The data has a lot of background information which is not useful in the lip reading task. We use the face-detector module in OpenCV to detect and extract faces from the images. This is important because our dataset is small, and we cannot afford the algorithm to waste computations on irrelevant parts of the image. After this step the size of each image becomes 90 X 90 X 3. This is not the final size of image passed for training, since different methods use different size by cropping it further as required. The cropped images are saved in separate folders called Training and Validation containing subfolders for the various class labels. Word utterances of speakers M07 and M08 were considered to be the validation set.

# **Data Augmentation:** Due to the small size of our dataset we perform data augmentation to artificially increase the data size [4]. Our data augmentation includes the following two modifications to the original image. While cropping, slightly move around the crop region by random number of pixels horizontally and vertically or Jitter the image by randomly increasing or decreasing the pixel values of the image by a small amount. Augmented data was generated using the ImageDataGenerator from the Keras package. The initial data set had 13726 images belonging to 10 classes in the train set and 1765 images in the validation set.

# **The Model:** We use the transfer learning techniques to increase the efficiency and reduce compute time.Transfer learning involves reusing a previously constructed model architecture and most of the learned weights, and then using standard training methods to learn the remaining, non-reused parameters. A fully trained neural net takes input values in an initial layer and then sequentially feeds this information forward (while simultaneously transforming it) until, crucially, some second-to-last layer has constructed a high level representation of the input that can more easily be transformed into a final output. The full training of the model involves the optimization of weight and bias terms used in each connection.The second-to-last layer is referred to as a bottleneck layer. The bottleneck layer pushes values in a regression model, or softmax probabilities in a classification model, to our final network layer. We have used the VGG16 architecture, that is pre-trained on the ImageNet dataset to find our bottleneck layer. Next we have trained a small fully-connected network (the top model) using the bottleneck features as input, with our classes as the classifier output.

# **Generating Predictions:** In order to predict the class of an image, we need to run it through the same pipeline as before. We first run the image through the pretrained VGG16 model (without the fully-connected layers again) and get the bottleneck predictions. We then run the bottleneck prediction through the trained top model - which we created in the previous step and get the final classification.

# **Camera for real time lip reading:** The video stream is captured by the webcam using the FPS and CV2 modules in python. The frames are grabbed from the stream and resized to have a maximum width of 400 pixels. The captured frames are preprocessed and passes through the same set of steps as performed on the validation set. A predicted word is obtained for each frame. The word having the maximum probability among a set of frames is considered to be the predicted word.

# **ALGORITHM FLOWCHART**

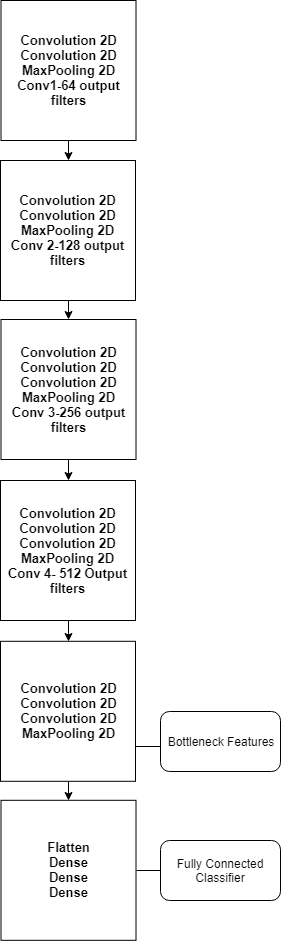


Figure 1:VGG-16 model architecture with modified top layer

# **IMPLEMENTATION AND RESULT**

The MIRACL-VC1 data set [7] containing both depth and color images of fifteen speakers uttering ten words and ten phrases, ten times each was used. The sequence of images represents low quality video frames. The data set contains 3000 sequences of varying lengths of images of 640 x 480 pixels, in both color and depth representations, collected at 15 frames per second. The lengths of these sequences range from 4 to 27 image frames. The words and phrases are as follows:

* Words: begin, choose, connection, navigation, next, previous, start, stop, hello, web
* Phrases: Stop navigation, Excuse me, I am sorry, Thank you, Good bye, I love this game, Nice to meet you, You are welcome, How are you, Have a good time.

To utilize time and lessen the size of the data, we focused on building a classifier that can identify which word is being uttered from a sequence of images of the speaker as input. The set of phrase data and also the depth of the images for the spoken word data was ignored and classifiers were built for both seen and unseen people. Where, for seen people the model is trained on all people but some trials are saved for testing and validation. In unseen, the people in train, test and validation are exclusive. The split is thirteen people for train, one for validation, and one for test. The resulting datasets are examples for unseen. The class label distribution for the dataset is even, as each person performs the same number of trials per word. Preprocessing was an important part of working with this dataset. First, we utilized a python facial recognition library, dlib, in conjunction with OpenCV and a pre-trained model [2] to isolate the points of facial structure in each image and crop it to only include the face of the speaker, excluding any background that could interfere with the training of the model.

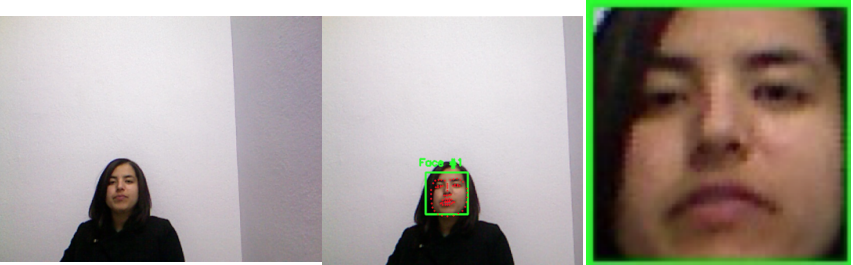


Figure 2: (left to right) Original Input image (part of a sequence) in the MIRACL-VC dataset; OpenCV and dlib facial recognition software labelling key points on around a detected face; final cropped image.

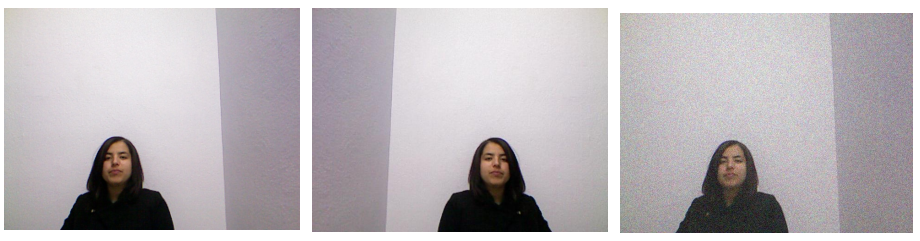


Figure 3: (left to right) Original Input image (part of a sequence) in the MIRACL-VC dataset; a horizontally flipped image; a jittered image.

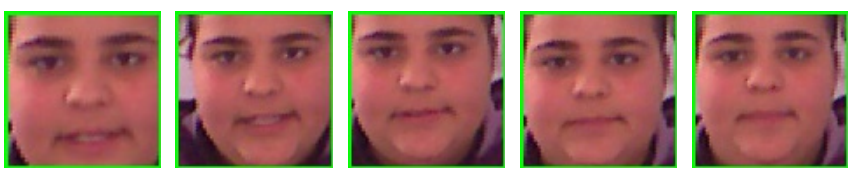


Figure 4: Example full input sequence of length 5. The subject is speaking “begin”.

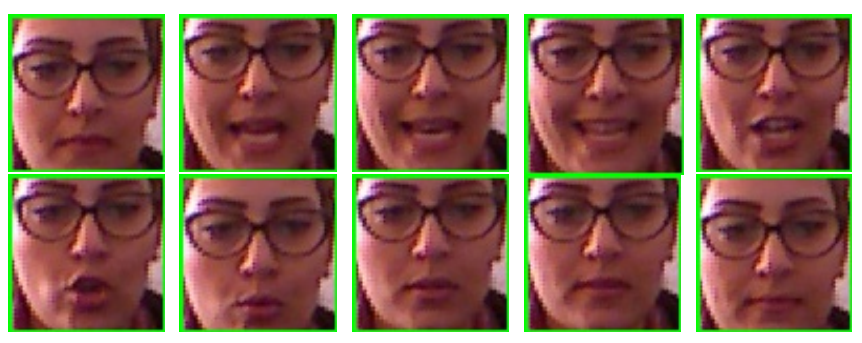
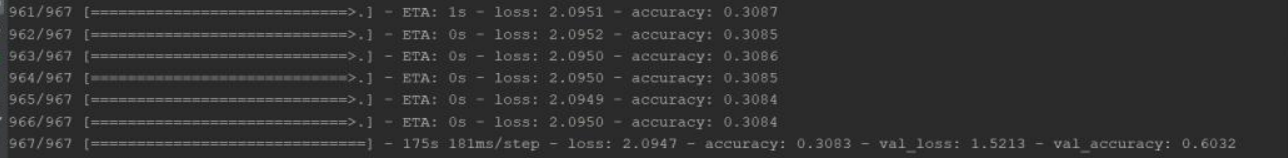


Figure 5: Example full input sequence of length 10. The subject is speaking “hello”

We had to limit the size of every facial crop to a 90x90x3 pixel square in order to create uniform input data sequences for the model.

After fitting our model with the training dataset and running 50 epoches,which takes about 30 minutes we obtain a training accuracy of 30.83% with loss as and a validation accuracy of 60.32%.



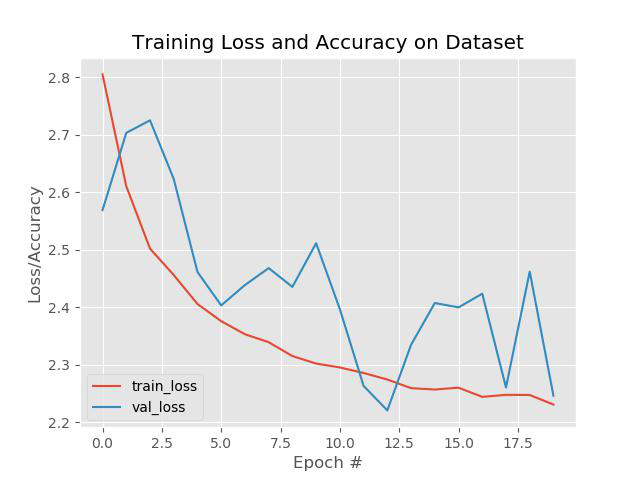
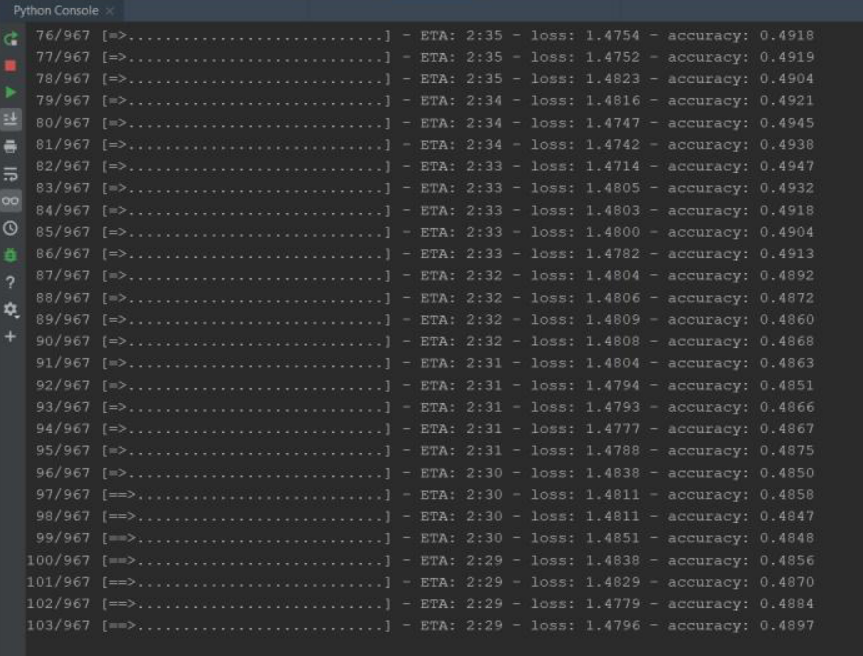


Figure 6: Training of the data and Loss on Training and Validation set.

We used accuracy as our primary metric, although we also looked at the recall rate of each class in the model to better understand where the errors were occurring. With 10 classes, a random baseline for this classifier is 0.1. The word “Start” has the maximum recall rate of 0.44.

We tested on both seen and unseen subjects. Results for seen subjects were relatively good, but our accuracy for unseen subjects gravitated barely above the random choice metric of 10% for all ten models. Our model is predicting “Start” for 92% of the words. We realized cross-validation could have helped mitigate this issue; a possible explanation for this result is that the person in the test set spoke faster than any other subject, and as a result, most of the words uttered by the subject are thought to be “Start”, since “Start” has perhaps the shortest pronunciation within the dataset.

Live implementation using webcam for the word “start” gives the following output:

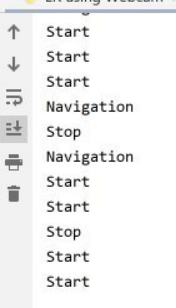
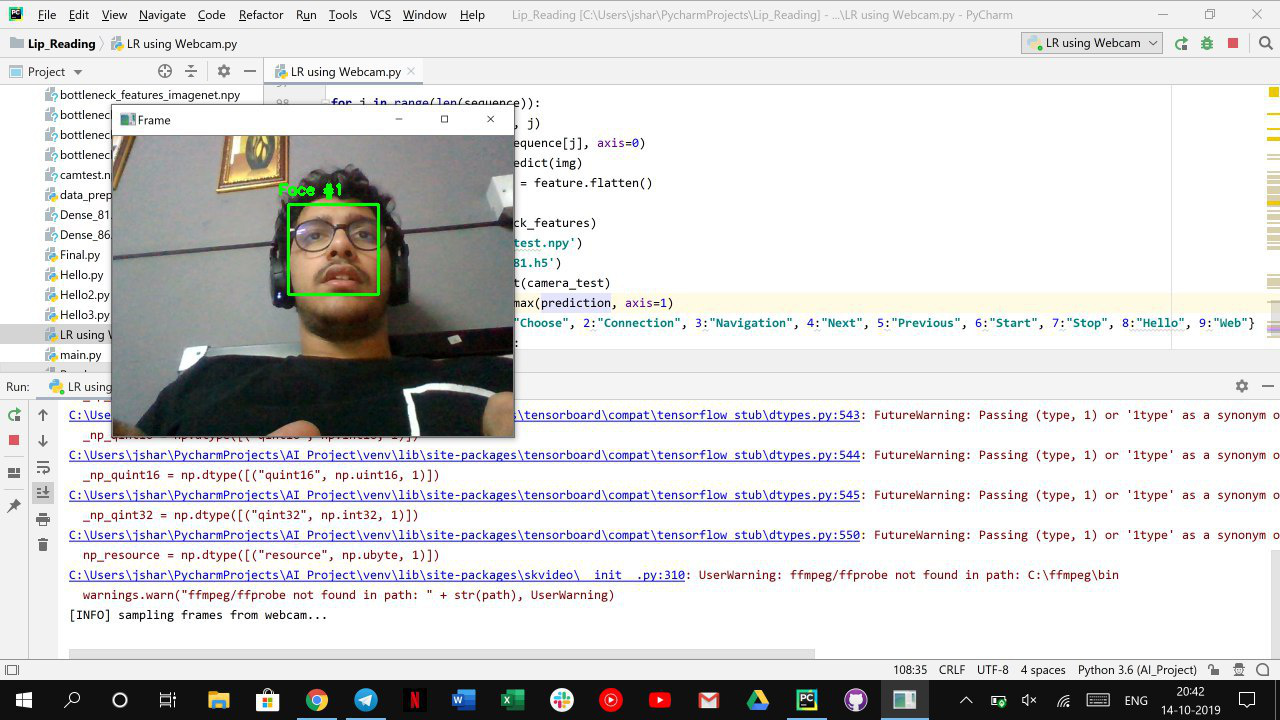


Figure 7: Live Implementation for the words.

# 

# **CONCLUSION AND RECOMMENDATION**

Overall, we found that the inclusion of pre-trained facial recognition CNNs highly improved our models. The augmentation of our data proved helpful but only in the instance of unseen people. Our best model had a validation accuracy of 60%. Generally, in all models, we found it very difficult to avoid overfitting with unseen people. Thus, certain models and hyperparameters are a better fit depending on whether we are working with seen or unseen people for testing and validation. More work needs to be done to reduce overfitting even seen people for the models that include pretrained networks.

Given more time and resources, the models outlined in this paper could be greatly improved.We also have yet to experiment with the number of filters in the fully connected layers. Use of LSTM or a TimeDistributed model could also be used to increase the accuracy. Additionally, accuracy improvements could be found with further hyperparameter tuning and investigation of even more optimizer types. We also would have gotten value from saliency maps. Without them it is hard to know if the model is accurately focusing on mouth data or other aspects of the input sequences.This project is easily extendible and raises the question of how to perform visual speech recognition on a much larger corpus (perhaps the entire English dictionary). How could the addition of audio data improve our ability to interpret the video as text? Is it easier to understand speech from video of a single word being spoken or entire phrases and sentences? This question could easily be investigated since the MIRACL-VC1 dataset includes phrase inputs and would be an interesting area of exploration. Additionally, most speech recognition tasks in real life require phrase inputs over single words.

1. **REFERENCES**

[1] Jaimes, A.; Sebe, N. Multimodal human–computer interaction: A survey. Comput. Vis. Image Underst. 2007, 108, 116–134

[2] Y. M. Assael, B. Shillingford, S. Whiteson, and N. de Freitas. “Lipnet: End-To-End Sentence-Level Lipreading”, arXiv preprint arXiv:1611.01599, 2016.

[3] McClain et al., 2004, Sumby and Pollack, 1954

[4] Amit Garg, Jonathan Noyola, Sameep Bagadia. “Lip reading using CNN and LSTM”

[5] Fatemeh Vakhshiteh, Farshad Almasganj, Ahmad Nickabadi, “Lip-Reading Via Deep Neural Networks Using Hybrid Visual Features”, 2018.

[6] Yannis M.Assael1, Brendan Shillingford1, ShimonWhiteson, NandodeFreita, “Lipnet: End-To-End Sentence-Level Lipreading”, 2017.

[7] Ahmed Rekik, Achraf BenHamadou, and Walid Mahdi. A new visual speech recognition approach for RGB-D cameras. In Image Analysis and Recognition - 11th International Conference, ICIAR 2014, Vilamoura, Portugal, October 22-24, 2014, Proceedings, Part II, pages 21–28, 2014.