LIP READING USING NEURAL NETWORK AND DEEP LEARNING

V. Sarala[1], **Chittiboina Leela Manusha** [2]

Assistant Professor MCA, DEPT, Dantuluri Narayana Raju College, **Bhimavaram, Andhra Pradesh**
Email id:- vedalasarala21@gmail.com
PG Student of MCA, Dantuluri Narayana Raju College, **Bhimavaram, Andhra Pradesh**
Email id:- leelamanusha111@gmail.com

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 task is difficult as people use different dictions and various ways to articulate a speech. 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 two separate CNN architectures. The trained lip reading models were evaluated based on their accuracy to predict words. The best performing model was implemented in a web application for real-time word prediction.

1 INTRODUCTION

Lip reading is a recent topic which has been a problematic concern to even expert lip readers. There is a scope for lip reading to be resolved using various methods of machine learning. Lip reading is a skill with salient benefits. Enhancement in lip reading technology increases the possibility to allow better speech recognition in noisy or loud environments. A prominent benefit would be developments in hearing aid systems 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 the audio is corrupted or absent in the video.

With the variety of languages spoken around the world, the difference in diction and relative articulation of words and phrases. It becomes substantially challenging to create a computer program that automatically and accurately reads the spoken words solely based on the visual lip movement of the speaker. Even the expert lip readers are only able to estimate about every second word [7]. Thus, utilizing the capabilities of neural networks and deep learning algorithms two architectures were trained and evaluated. Based on the evaluation, the better performing model was further customized to enhanced accuracy. The model architecture with an overall better accuracy was implemented in a web application to devise a realtime lip-reading system.

2.LITERATURE SURVEY AND RELATED WORK

A. Word Spotting in Silent Lip videos(Abhishek Jha, 2018) They have introduced pipeline which is recognition free retrieval for word spotting. They have used WAS and CMT lipreading model-based characteristic for word spotting in LRW dataset which have showed about 36%, 50% improvement across the recognition. Re-ranking method has been included additionally in the pipeline to increase the results of the retrieval. They have showed increment of 106% and 195% in domain uniformity of their pipeline. They have attained 35% greater average accuracy across recognition-based techniques.

B. Lip Reading Word Classification (Abiel Gutierrez, 2017) Their best model was the Fine-Tuned VGG+LSTM baseline. Data augmentation proved to be helpful only in instance of unseen people. Their baseline outperforms LSTM+CNN architecture. They achieved validation accuracy very close to 75% and test accuracy of 59%.

C. Lipnet: End to End Sentence Level Lip reading (Yannis M Assael, 2016) Their model Lipnet achieved 95.2%

accuracy in sentence level lipreading over human lipreaders. This model helps in eliminating the need of segmenting the videos in to words before predicting a sentence. They have proposed a very first model of lipnet which apply deep learning techniques to entire learning of model which maps the series of the images trained from speaker's mouth to whole sentences.

D. Lip Reading with Long ShortTerm Memory (Michael Wand, 2016) This paper reported a best word accuracy of 79.6% on held-out speakers. They have showed greater accuracy in the word by using the neural network based lipreading system than the system with pipeline using feature extraction and classification. 80% accuracy in the word from speaker-dependent lipreading has attained by using lipreading with single feed-forward network.

E. Lip Vison A Deep Learning Appraoch (Parth Khetarpal, 2017) They have discussed different techniques for lip and face detection and various classification techniques have been used, this is considered as their objective. Some of the features identified by them includes edges of lip, height and width of lips and angle between particular lip point and they have given the best accuracy of 88.6% over unseen speaker's by using the methods of CNN and RNN. The algorithms which have been proposed by them was tested with both speaker dependent and independent data which have given accurate recognition result though limited training data is available.

F. Lipnet: A Comparitive Study (Vyom Jain, 2017) The task of understanding the narration from the speaker's lip movement is called lipreading. Lipreading is considered as tough task for humans, mainly in the absence of subtitles. In this paper, they have discussed few approaches which have overcome the human difficulties. Their comparative study on lipreading has assisted us with well-known technologies and also to obtain a finer idea of the problem handy.

3 PROPOSED WORK AND ALGORITHM

Our goal for this project is to design an autonomous Lip Reading system to translate lip movements in real-time to coherent sentences. We will use deep learning to classify lip movements in the form of video frames to phonemes. Afterward, we stitch the phonemes into words and combine these words into sentences.

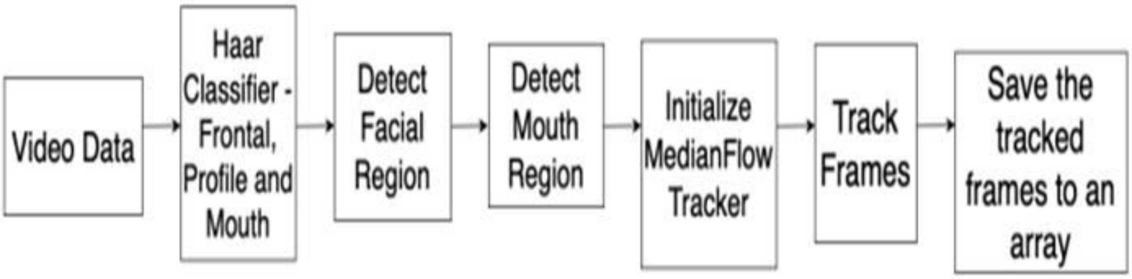

Video Data
Haar Classifier - Frontal, Profile and Mouth
Detect Facial Region
Detect Mouth Region
Initialize MedianFlow Tracker
Track Frames
Save the tracked frames to an array

Fig-1: System Architecture

4 METHODOLOGIES

INPUT DATASET: Using this module we will upload LRW dataset

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 and mouth region from each input videos. OpenCV is an open-source multi-platform library of programming functions focused on computer vision

TRAINING:

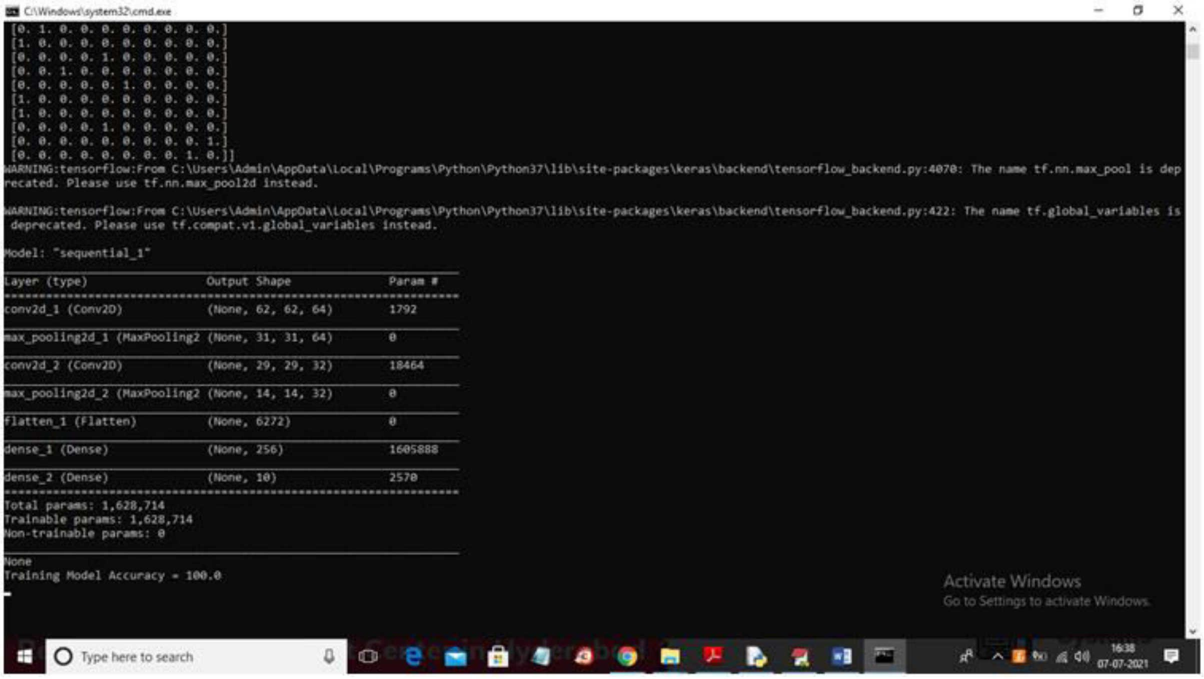
The lip reading model was trained with a CNN architecture. A 3-Dimensional (3D) CNN was applied to train the pre-processed lip samples and compare various parameters. Due to the 3D CNN architecture' scalability of training with high dimensional data like image sequences

Generate CNN Model:Using this module we can see CNN modelis generated

EVALUATION:

In the field of machine learning, evaluation of the model is a significant task. It is important to know if the trained model has learned patterns to generalize the prediction in unseen data to avoid overfitting on the lip reading model. For measuring the predictive accuracy of the model I performed a Top-1 accuracy

5.RESULTSANDDISCUSSION SCREENSHOTS



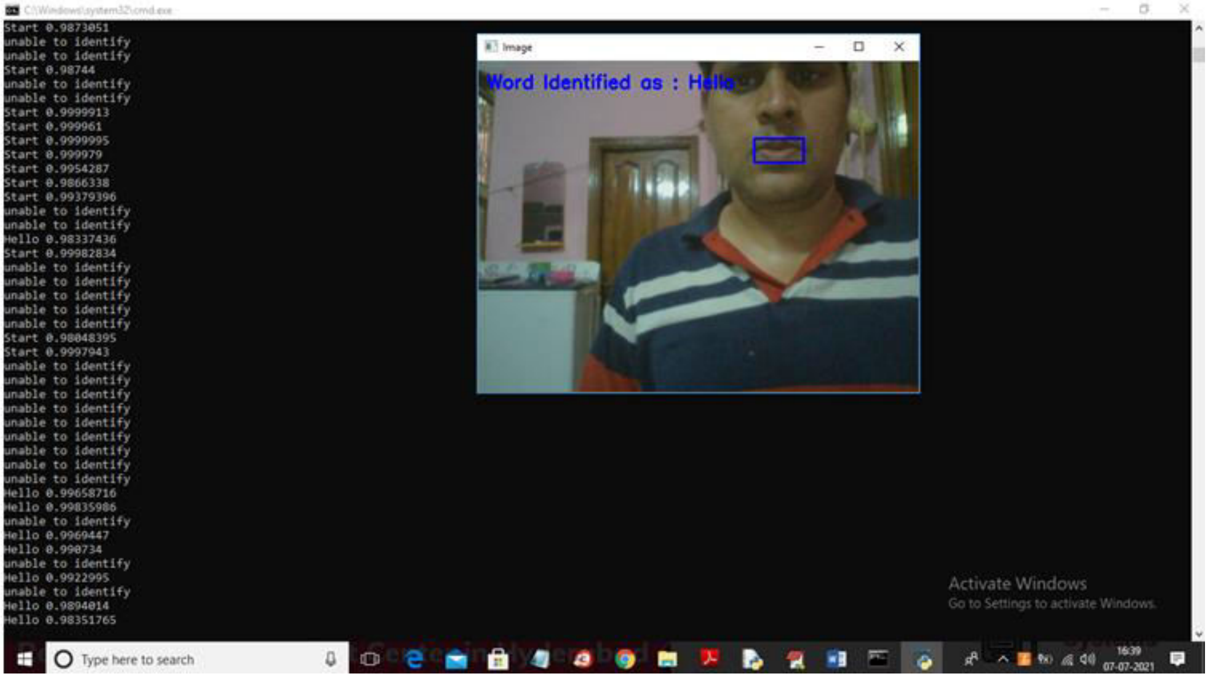
C:\Windows\system32\cmd.exe
Activate Windows
Go to Settings to activate Windows.
Type here to search
16:38
07-07-2021

```
 [0. 1. 0. 0. 0. 0. 0. 0. 0. 0.]
 [1. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 1. 0. 0. 0. 0. 0.]
 [0. 0. 1. 0. 0. 0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 0. 1. 0. 0. 0. 0.]
 [1. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
 [1. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 1. 0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
 [0. 0. 0. 0. 0. 0. 0. 0. 1. 0.]]
WARNING:tensorflow:From C:\Users\Admin\AppData\Local\Programs\Python\Python37\lib\site-packages\keras\backend\tensorflow_backend.py:4070: The name tf.nn.max_pool is dep
recated. Please use tf.nn.max_pool2d instead.

WARNING:tensorflow:From C:\Users\Admin\AppData\Local\Programs\Python\Python37\lib\site-packages\keras\backend\tensorflow_backend.py:422: The name tf.global_variables is
 deprecated. Please use tf.compat.v1.global_variables instead.

Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 62, 62, 64)        1792
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 31, 31, 64)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 29, 29, 32)        18464
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 14, 14, 32)        0
_________________________________________________________________
flatten_1 (Flatten)          (None, 6272)              0
_________________________________________________________________
dense_1 (Dense)              (None, 256)               1605888
_________________________________________________________________
dense_2 (Dense)              (None, 10)                2570
=================================================================
Total params: 1,628,714
Trainable params: 1,628,714
Non-trainable params: 0
_________________________________________________________________
None
Training Model Accuracy = 100.0
```

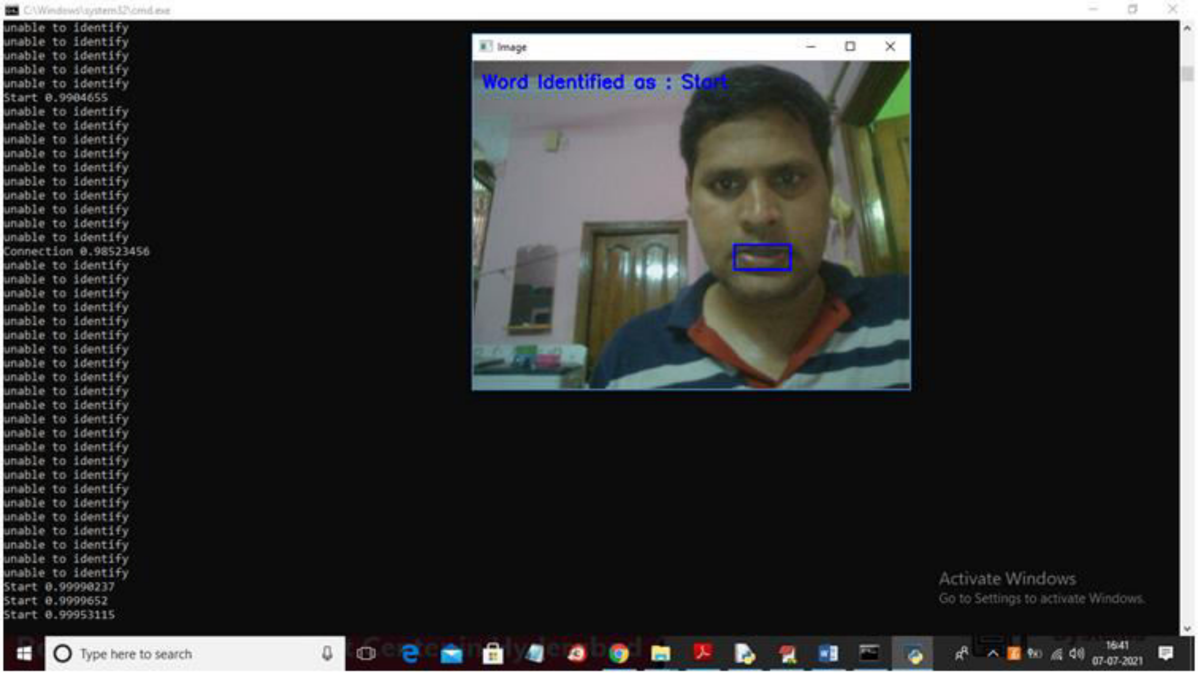
Fig 2:- In above screen CNN model training is done on dataset and we got 100% accuracy



C:\Windows\system32\cmd.exe
Image
Word Identified as : Hello
Activate Windows
Go to Settings to activate Windows.
Type here to search
16:39
07-07-2021

```
Start 0.9873051
unable to identify
unable to identify
Start 0.98744
unable to identify
unable to identify
Start 0.9999913
Start 0.999961
Start 0.9999995
Start 0.999979
Start 0.9954287
Start 0.9866338
Start 0.99379396
unable to identify
unable to identify
Hello 0.98337436
Start 0.99982834
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
Start 0.98048395
Start 0.9997943
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
Hello 0.99658716
Hello 0.99835986
unable to identify
Hello 0.9969447
Hello 0.990734
unable to identify
Hello 0.9922995
unable to identify
Hello 0.9894014
Hello 0.98351765
```

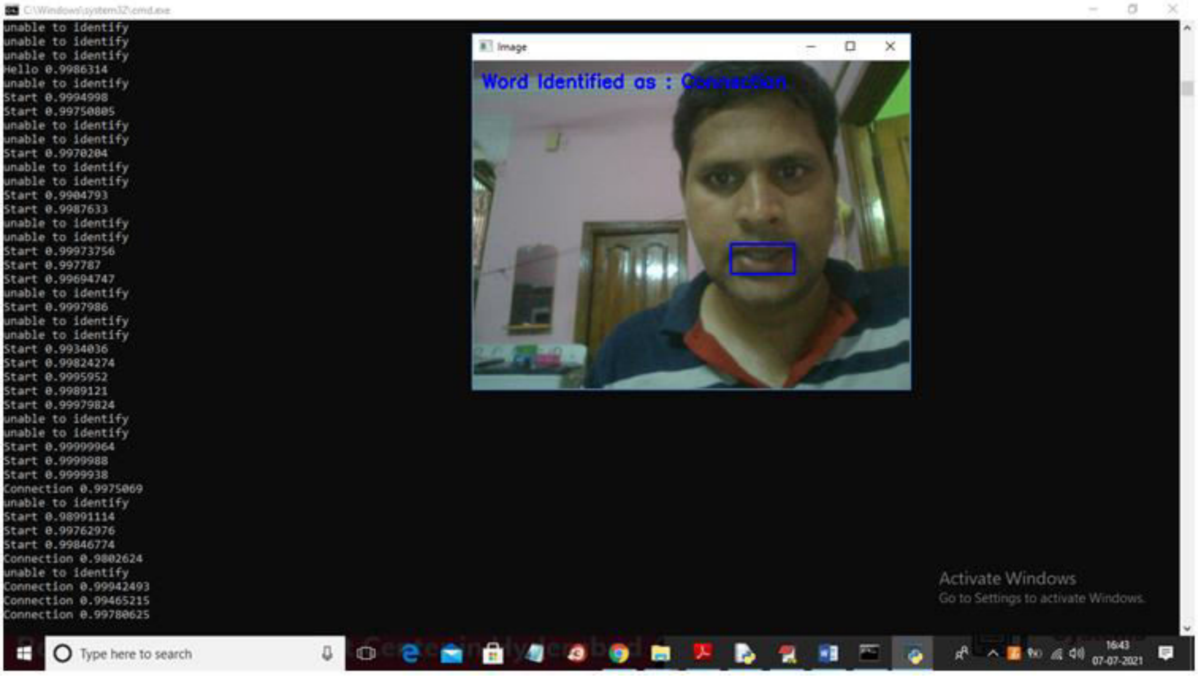
Fig 3:- output-1



C:\Windows\system32\cmd.exe
Image
Word Identified as : Start
Activate Windows
Go to Settings to activate Windows.
Type here to search
16:41
07-07-2021

```
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
Start 0.9904655
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
Connection 0.98523456
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
unable to identify
Start 0.99990237
Start 0.9999652
Start 0.99953115
```

Fig 4 :- output-2



C:\Windows\system32\cmd.exe
Image
Word Identified as : Connection
Activate Windows
Go to Settings to activate Windows.
Type here to search
16:43
07-07-2021

```
unable to identify
unable to identify
unable to identify
Hello 0.9986314
unable to identify
Start 0.9994998
Start 0.99750805
unable to identify
unable to identify
Start 0.9970204
unable to identify
unable to identify
Start 0.9904793
Start 0.9987633
unable to identify
unable to identify
Start 0.99973756
Start 0.997787
Start 0.99694747
unable to identify
Start 0.9997986
unable to identify
unable to identify
Start 0.9934036
Start 0.99824274
Start 0.9995952
Start 0.9989121
Start 0.99979824
unable to identify
unable to identify
Start 0.99999964
Start 0.9999988
Start 0.9999938
Connection 0.9975069
unable to identify
Start 0.98991114
Start 0.99762976
Start 0.99846774
Connection 0.9802624
unable to identify
Connection 0.99942493
Connection 0.99465215
Connection 0.99780625
```

Fig 5 :- output-3

6. CONCLUSION

To predict lip reading application will take images from webcam and then apply HAAR CASCADE files to detect face and mouth and then detected mouth will be input to CNN to identify word based on lips movement.

7. REFERENCES

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. 2016. Tensorflow: A system for large-scale machine learning. In 12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16). 265– 283.

[2] Gary Bradski and Adrian Kaehler. 2008. Learning OpenCV: Computer vision with the OpenCV library. " O'Reilly Media, Inc.".

[3] L. Chen. 2016. keras,js. https://github.com/transcranial/keras-js

[4] François Chollet. 2015. Keras documentation. keras. io (2015).

[5] Joon Son Chung and Andrew Zisserman. 2016. Lip reading in the wild. In Asian Conference on Computer Vision. Springer, 87–103.

[6] Dan Hammerstrom. 1993. Neural networks at work. IEEE spectrum 30, 6 (1993), 26–32.

[7] Ahmad BA Hassanat. 2011. Visual Speech Recognition, Speech and Language Technologies, Prof. Ivo Ipsic (Ed.), ISBN: 978-953-307-322-4, InTech.

[8] Simon Haykin. 1994. Neural networks: a comprehensive foundation. Prentice Hall PTR.

[9] Hynek Hermansky. 1990. Perceptual linear predictive (PLP) analysis of speech. the Journal of the Acoustical Society of America 87, 4 (1990), 1738–1752.

[10] Sanghoon Hong, Byungseok Roh, Kye-Hyeon Kim, Yeongjae Cheon, and Minje Park. 2016. Pvanet: Lightweight deep neural networks for real-time object detection. arXiv preprint arXiv:1611.08588 (2016).

[11] Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015).

[12] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014).

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1 (NIPS'12). Curran Associates Inc., USA, 1097–1105. http://dl.acm.org/citation. cfm?id=2999134.2999257

[14] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems. 1097–1105.

[15] Yiting Li, Yuki Takashima, Tetsuya Takiguchi, and Yasuo Ariki. 2016. Lip reading using a dynamic feature of lip images and convolutional neural networks. In 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS). IEEE, 1–6.

[16] Satya Mallick. 2017. Home. https://www.learnopencv.com/ object-tracking-using-opencv-cpp-python/

[17] Kuniaki Noda, Yuki Yamaguchi, Kazuhiro Nakadai, Hiroshi G Okuno, and Tetsuya Ogata. 2014. Lipreading using convolutional neural network. In Fifteenth Annual Conference of the International Speech Communication Association.

[18] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. Journal of machine learning research 12, Oct (2011), 2825–2830.

[19] Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, and Aleksander Madry. 2018. How does batch normalization help optimization?. In Advances in Neural Information Processing Systems. 2483–2493.

[20] Wenling Shang, Kihyuk Sohn, Diogo Almeida, and Honglak Lee. 2016. Understanding and improving convolutional neural networks via concatenated rectified linear units. In international conference on machine learning. 2217–2225.