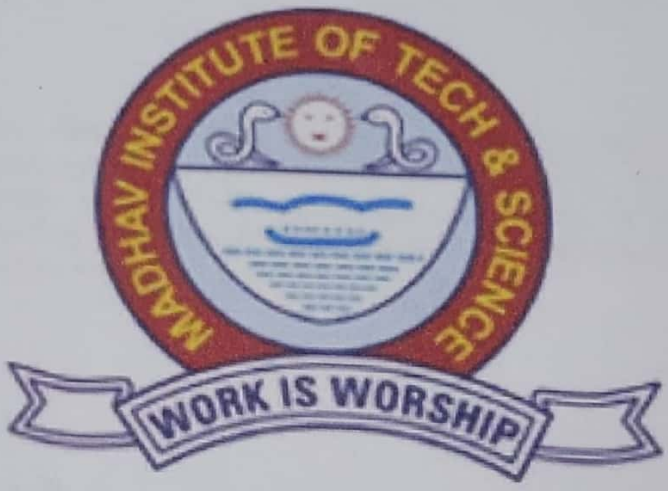
MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE GWALIOR
(A Govt. Aided UGC Autonomous Institute Affiliated to RGPV, Bhopal)
NAAC Accredited with A++ Grade



Project Report

On

LipNet

Submitted By:

Samriddhi Fuskelay (0901AM211049)

Vansh Gawra (0901AM211062)

Faculty Mentor:

Dr. Sunil Kumar Shukla

CENTRE FOR ARTIFICIAL INTELLIGENCE
MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE
GWALIOR - 474005 (MP) est. 1957

JULY-DEC, 2023

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE GWALIOR

(A Govt. Aided UGC Autonomous Institute Affiliated to RGPV, Bhopal)

NAAC Accredited with A++ Grade

CERTIFICATE

This is certified that **Samriddhi Fuskelay (0901AM211049), Vansh Gawra (0901AM211062)** has submitted the project report titled **LipNet** under the mentorship of **Dr. Sunil Kumar Shukla**, in partial fulfilment of the requirement for the award of degree of Bachelor of Technology in **Artificial Intelligence and Machine Learning** from Madhav Institute of Technology and Science, Gwalior.

23/11/2023

Dr. Sunil Kumar Shukla
Faculty Mentor
Assistant professor
Centre for Artificial Intelligence

23/11/23

Dr. R. R. Singh
Coordinator
Centre for Artificial Intelligence

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE GWALIOR

(A Govt. Aided UGC Autonomous Institute Affiliated to RGPV, Bhopal)

NAAC Accredited with A++ Grade

DECLARATION

I hereby declare that the work being presented in this project report, for the partial fulfilment of requirement for the award of the degree of Bachelor of Technology in **Artificial Intelligence and Machine Learning** at Madhav Institute of Technology & Science, Gwalior is an authenticated and original record of my work under the mentorship of **Dr. Sunil Kumar Shukla,** Assistant professor, Centre of Artificial Intelligence

I declare that I have not submitted the matter embodied in this report for the award of any degree or diploma anywhere else.

Samriddhi Fuskelay
0901AM211049
3rd Year,
Centre for Artificial Intelligence

Vansh Gawra
0901AM211062
3rd Year,
Centre for Artificial Intelligence

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE GWALIOR

(A Govt. Aided UGC Autonomous Institute Affiliated to RGPV, Bhopal)

NAAC Accredited with A++ Grade

ACKNOWLEDGEMENT

The full semester project has proved to be pivotal to my career. I am thankful to my institute, **Madhav Institute of Technology and Science** to allow me to continue my disciplinary/interdisciplinary project as a curriculum requirement, under the provisions of the Flexible Curriculum Scheme (based on the AICTE Model Curriculum 2018), approved by the Academic Council of the institute. I extend my gratitude to the Director of the institute, **Dr. R. K. Pandit** and Dean Academics, **Dr. Manjaree Pandit** for this.

I would sincerely like to thank my department, **Centre for Artificial Intelligence,** for allowing me to explore this project. I humbly thank **Dr. R. R. Singh**, Coordinator, Centre for Artificial Intelligence, for his continued support during the course of this engagement, which eased the process and formalities involved.

I am sincerely thankful to my faculty mentors. I am grateful to the guidance of **Dr. Sunil Kumar Shukla**, Assistant professor, Centre of Artificial Intelligence, for his continued support and guidance throughout the project. I am also very thankful to the faculty and staff of the department.

Samriddhi Fuskelay
0901AM211049
3rd Year,
Centre for Artificial Intelligence

Vansh Gawra
0901AM211062
3rd Year,
Centre for Artificial Intelligence

Table of Contents

ABSTRACT

The "LipNet" project comes from a shared passion for improving communication, especially for people who may face challenges in hearing. Our main goal is to use advanced technology, specifically 3D Convolutional Neural Networks (3D CNN) & Bi-Directional LSTM, to understand and interpret spoken words by analysing the movements of the lips. This contributes to the field of visual speech recognition.

To start, we utilized the "The Grid Audio-Visual Speech Corpus" dataset's subset that had 1000 samples, capturing different lip movements during speech. From this collection, 900 videos were used to teach our model, giving it a strong foundation. The remaining 100 videos were set aside to test how well the model can understand new, unseen lip movements.

Our approach followed a step-by-step process. First, we developed functions to organize and prepare the lip movement data efficiently. Then, we built a smart system, known as a data pipeline, to help our model understand and learn from the videos. This step is crucial for accurate lip reading.

After preparing the data, we created a hybrid neural network designed to learn from the videos. The model was trained on the 900-sample dataset, learning to recognize patterns in lip movements. Next, we tested the model on the reserved 100-sample dataset, using Average Word Accuracy to measure how well it performed.

The success of "LipNet" goes beyond just technology. It has the potential to make communication better for people with hearing difficulties by understanding spoken language through visual cues.

In summary, "LipNet" is a step forward in combining advanced technology with visual speech recognition. This summary covers everything from gathering and preparing the videos to training our model and testing its performance. The impact of this project goes beyond its immediate use, promising to improve communication for those who rely on visual cues in understanding spoken words.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviation	Description
3D CNN	3D Convolutional Neural Network
LSTM	Long Short-Term Memory
ML	Machine Learning
STCNN	Spatiotemporal Convolutional Neural Network
RNN	Recurrent Neural Network
KD	Knowledge Distillation
VSD	Visual Speech Detection
MVM	Multi-head Visual-audio Memory
ALR	Automatic Lip Reading
DL	Deep Learning
EVWF	Enhanced Visually-Derived Wiener Filter
SS	Spectral Subtraction
LMMSE	Log-Minimum Mean-Square Error
CTC	Connectionist Temporal Classification

Chapter 1: PROJECT OVERVIEW

1.1. Introduction

In the big world of talking and understanding, "LipNet" is a project that wants to make communication better, especially for people who might find it hard to hear. It's all about using smart technology, like computers that can learn, to understand what people are saying just by looking at how their lips move. "LipNet" uses 3D Convolutional Neural Networks and Bi-Directional LSTM and Deep learning principles, which are like super-smart tools, to do this.

1.1.1. Project Genesis:

"LipNet" began because we really want to help people communicate better, especially if they use lip movements to understand what others are saying and installing lipreading models in security sector. We use cool technology to teach the computer to understand spoken words by watching lip movements closely. The focus is on making communication easier, especially for those who might have trouble hearing.

1.1.2. Significance:

In a world where technology is always changing how we do things, "LipNet" is important because it helps computers understand spoken language using visual cues. This can make talking and understanding easier, especially for people who rely on visual signals to get the message.

1.1.3. Scope:

Beyond the realm of lip reading, "LipNet" delves into the broader intersection of technology and communication. By employing smart techniques and tools, the project explores innovative ways of understanding and facilitating communication. This exploration opens doors to numerous applications, such as:

1. **Enhanced Accessibility**: Providing a valuable tool for individuals with hearing impairments to engage in seamless communication.
2. **Security and Surveillance**: Contributing to security measures by potentially aiding in crime prevention through lip reading technology.
3. **Multimodal Interaction**: Beyond lip reading, "LipNet" explores innovative ways to facilitate communication, contributing to the field of human-computer interaction and enabling more nuanced applications.

1.2. Objectives and Scope

1.2.1. Project Objectives:

The "LipNet" project is designed with a diverse set of objectives, aiming to harmoniously blend technological innovation with the enhancement of communication accessibility. Our specific goals include:

- **Lip Reading Precision:** Develop a robust system capable of accurately deciphering spoken words through lip movements, contributing to the field of visual speech recognition.

- **Deep Learning Integration:** Harness the power of 3D Convolutional Neural Networks (3D CNNs) to create a deep learning model proficient in understanding temporal features of lip movements for precise speech interpretation.

- **Dataset Selection:** Chose a dataset comprising 3-second videos, ensuring varied lip movements and high-quality visuals for effective training and testing.

- **Model Training and Evaluation:** Implement a systematic training pipeline for the deep learning model, evaluating its accuracy and performance across diverse datasets.

1.2.2. Project Scope:

The scope of the project extends to:

- **Lip Movement Variability:**Encompass a wide range of lip movements to enhance the model's versatility in recognizing diverse spoken words and expressions.

- **Real-world Simulation**: Integrate real-world elements, such as background noise and varying environments, into the dataset to enhance the model's adaptability to practical scenarios.

- **Application Testing:** Apply the trained model to interpret spoken words within recorded conversations, demonstrating the practical utility and real-world application of the system.

1.2.3. Expected Outcomes:

Through the successful realization of these objectives, we anticipate achieving the following outcomes:

- **Accurate Speech Decoding:** Attain a high level of accuracy in decoding spoken words from lip movements, contributing to the advancement of visual speech recognition technology.
- **Model Adaptability**: Develop a model capable of adapting its learnings to effectively understand a variety of spoken expressions and communication nuances.
- **Contribution to Communication Technology:** Contribute insights and advancements to the intersection of technology and communication, opening new possibilities for inclusive communication and accessibility.

1.3. Project Features

1.3.1. Lip reading precision:

A defining feature of the "LipNet" project is its capacity to precisely decipher spoken words through lip movements. The project employs advanced techniques, specifically 3D Convolutional Neural Networks (3D CNNs) & Bi-Directional LSTM, to interpret and categorize the intricate temporal features within video data.

1.3.2. Deep Learning Integration:

By integrating TensorFlow's Keras, "LipNet" seamlessly incorporates deep learning methodologies into the realm of visual speech recognition. The adoption of 3D CNNs & Bi-Directional LSTM facilitates the extraction of nuanced features from lip movement data, enhancing the model's proficiency in understanding variations in spoken language..

1.3.3. Dataset Selection:

To ensure the adaptability and effectiveness of the model. The dataset comprises 3-second videos, offering a diverse range of lip movements for both training and testing. Real-world elements, such as background noise and varying environments, were deliberately included to simulate practical scenarios.

1.3.4. Model Training and Evaluation:

The project features a comprehensive model training pipeline, involving the systematic partitioning of data for training and testing. The model undergoes training over multiple epochs, incorporating an early stopping mechanism. The evaluation phase rigorously assesses the model's accuracy and its ability to generalize to a variety of spoken expressions.

1.3.5. Real-world Applicability:

Beyond lip reading precision, "LipNet" extends its features to real-world scenarios. The deliberate inclusion of real-world elements in the dataset ensures that the model is not only technically feasible but also practically applicable in authentic contexts, enhancing its adaptability and reliability.

1.4. Feasibility

1.4.1. Technical Feasibility:

The "LipNet" project is technically feasible, leveraging advanced technologies like 3D CNNs for visual speech recognition. TensorFlow's Keras provides a reliable framework for deep learning, and the feasibility is substantiated by successful applications in similar domains.

1.4.2. Dataset Collection and Preprocessing:

Feasibility is addressed through the meticulous collection and preprocessing of the dataset. The dataset's design, with diverse lip movements and real-world elements, ensures it represents authentic scenarios, contributing to the model's effectiveness.

1.4.3. Model Training and Evaluation:

Feasibility is rigorously tested during the model training and evaluation phases. The model architecture, with 3D CNN layers, max pooling, time distribution, bidirectional LSTM is configured for effective feature extraction from lip movement data. Evaluation ensures high accuracy and generalization capabilities across varied spoken expressions.

1.4.4. Real-world Applicability:

The project's feasibility extends to real-world scenarios, with intentional inclusion of real-world elements in the dataset. This ensures the model is not only technically feasible but also practically applicable, enhancing its reliability in real-life usage.

1.5. System Requirements

1.5.1. Hardware Requirements:

The "LipNet" project operates within reasonable hardware specifications for accessibility and efficiency. Recommended requirements include:

- **Processor**: Quad-core processor or higher for efficient data processing.
- **Memory (RAM):** 8 GB or more to handle computational demands.
- **Storage:** Adequate capacity for dataset and model files.
-

1.5.2. Software Requirements:

To facilitate development and execution, the project relies on key software components:

- **Python:** Primary language for implementation.
- **TensorFlow with Keras**: For developing and training the 3D CNN model.
- **Imageio**: an easy interface to read and write a wide range of image data.
- **OpenCV:** OpenCV provides a real-time optimized Computer Vision library, tools, and hardware. It also supports model execution for Machine Learning (ML).
- **Matplotlib**: Employed for visualizations.

Chapter 2: LITERATURE REVIEW

2.1. Lip Reading

2.1.1. Historical Context:

The challenge of lip reading in communication has deep historical roots, initially relying on manual interpretation. Early methods were limited by the manual transcription of lip movements, impeding the scalability and speed of lip reading analysis. Technological advancements have paved the way for computational approaches, introducing speed and precision to the process.

2.1.2. Advancements in Automated Lip Reading:

The literature unfolds a progression from manual interpretation to automated lip reading systems. Rule-based systems faced challenges in handling the variability of speech movements. Recent strides in technology showcase the dominance of deep learning, particularly 3D Convolutional Neural Networks (3D CNNs) & Bi-Directional LSTMs, for improved lip reading average word accuracy.

2.1.3. Relevance of Lip Reading:

Automated lip reading holds significance in diverse domains, from enhancing communication accessibility to security and surveillance applications. The computational identification of spoken words through lip movements opens avenues for improved communication for individuals with hearing impairments and contributes to advancements in human-computer interaction.

2.1.4. Challenges and Open Problems:

Despite technological advancements, universal lip reading remains a challenge, especially in real-world scenarios with varying expressions and environmental conditions. The intricacies of human lip movements, coupled with diverse accents and expressions, existence of HOMOPHONES present ongoing research opportunities..

2.1.5. Integration of 3D CNN:

Recent studies highlight the successful integration of 3D CNNs for lip reading. These deep learning models demonstrate a capacity to learn temporal features within video data, making them well-suited for the complexities of lip movement analysis.

2.1.6. Integration of Bi-Directional LSTM:

Integration of Bi-Directional LSTMs enhances lip reading models alongside 3D CNNs. Bi-LSTMs excel in capturing temporal dynamics, crucial for interpreting lip movements. Their bidirectional processing considers both past and future frames, improving overall temporal context. This synergistic approach enhances accuracy in recognizing spoken words, advancing applications in human-computer interaction and speech recognition.

2.2. Existing Approaches and Technologies

2.2.1. Manual Interpretation:

Early lip reading attempts relied on manual interpretation, posing challenges in scalability and accuracy. Manual decoding of lip movements struggled to handle the diversity of expressions and real-world conditions.

Fig 2.2.1 – Manual Interpretation

2.2.2. Rule-Based Systems:

Early automated systems encoded predefined rules based on lip movement patterns. However, they faced limitations in handling the nuances of natural communication and struggled with real-world variability.

2.2.3. Machine Learning-Based Approaches:

The evolution of machine learning marked a transformative shift in lip reading. The adoption of 3D CNNs showcased improved accuracy in deciphering spoken words through lip movements, reducing reliance on manual decoding.

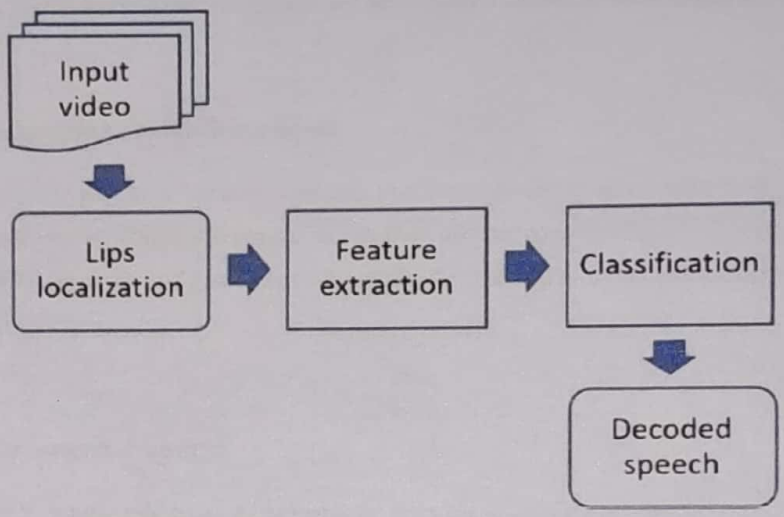


Fig 2.2.3 – Machine learning based approach in lip reading.

2.3. Relevance of Deep learning in Communication Analysis

2.3.1. Automated Feature Learning:

Depe learning, particularly 3D CNNs & Bi-Directional LSTMs, has automated the feature learning process in lip reading. These models excel at extracting temporal features from video data, streamlining the interpretation of spoken words through lip movements.

2.3.2. Enhanced Accuracy and Generalization:

Deep learning's application in lip reading has led to enhanced accuracy and generalization. Models trained on diverse datasets can recognize spoken words across various accents and expressions, adapting to different communication styles.

2.3.3. Real-time Analysis and Recommendations:

Deep learning facilitates real-time lip reading analysis, offering instant feedback during live communication or video streaming. Recommendation systems leveraging machine learning algorithms can contribute to more accessible and personalized communication experiences.

2.3.4. Challenges and Future Directions:

While Deep learning has propelled advancements, challenges persist in achieving robust lip reading across diverse scenarios. Ongoing research may explore unsupervised learning for lip movement discovery and the integration of contextual information for nuanced analyses in real-world communication scenarios.

2.4. Past Research Papers:

2.4.1. LIPNET: END-TO-END SENTENCE-LEVEL LIPREADING

The main objective of this research is to address the challenging task of lipreading, specifically focusing on decoding text from the movement of a speaker's mouth. The traditional approach, which involves separating the problem into designing or learning visual features and prediction stages, is compared with more recent end-to-end trainable deep lipreading approaches. The paper introduces LipNet, a novel model designed to map a variable-length sequence of video frames to text.

LIPNET:

- LipNet is presented as the first end-to-end sentence-level lipreading model.
- Utilizes spatiotemporal convolutional neural networks (STCNNs), recurrent neural networks (RNNs), and the connectionist temporal classification loss for character-level predictions.
- Trained end-to-end for sentence-level lipreading, in contrast to existing approaches focusing on word classification.

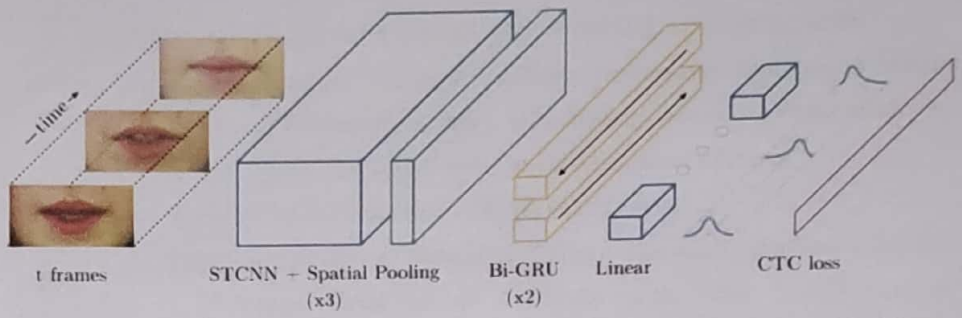


Figure 1: LipNet architecture. A sequence of T frames is used as input, and is processed by 3 layers of STCNN, each followed by a spatial max-pooling layer. The features extracted are processed by 2 Bi-GRUs; each time-step of the GRU output is processed by a linear layer and a softmax. This end-to-end model is trained with CTC.

2.4.2. HEARING LIPS: IMPROVING LIP READING BY DISTILLING SPEECH RECOGNIZERS (2020)

The paper addresses the challenge of improving lip reading performance, which lags behind speech recognition, despite recent advancements in deep learning and large-scale datasets.

METHODOLOGY:

- **KNOWLEDGE DISTILLATION (KD):**
  The core idea is to distill knowledge from speech recognizers to enhance lip reading, acknowledging the complementary information present in acoustic speech signals.
  Multi-Granularity Knowledge Distillation: LIBS distills knowledge at multiple temporal scales, including sequence-level, context-level, and frame-level.

- **CROSS-MODAL ALIGNMENT SCHEME:**
  Handling Asynchronous Data: LIBS employs an efficient alignment scheme to handle inconsistent lengths of audio and video data, addressing the challenge posed by asynchronous modalities. This synchronization is crucial for fine-grained knowledge distillation.
- **INNOVATIVE FILTERING STRATEGY:**
  Refining Speech Recognizer's Predictions: Acknowledging imperfect speech recognition predictions, LIBS introduces an innovative filtering strategy to refine the features distilled from the speech recognizer.

2.4.3. DEEP LEARNING-BASED AUTOMATED LIP-READING: A SURVEY

- **DEEP LEARNING FOR FEATURE EXTRACTION AND CLASSIFICATION:**
  Inclusion of Comparative Analysis: The paper includes comparative analyses of various components within automated lip-reading systems. This encompasses audio-visual databases, feature extraction methods, classification networks, and classification schemas.
- **COMPARISON OF NEURAL NETWORK ARCHITECTURES:**
  1. **Focus on Convolutional Neural Networks (CNNs):** The survey provides a detailed comparison of Convolutional Neural Networks with other neural network architectures concerning feature extraction in the context of lip reading.
  2. **Identification of Advantages:** The paper critically reviews the advantages of Attention-Transformers and Temporal Convolutional Networks, particularly in comparison to Recurrent Neural Networks (RNNs) for classification tasks in lip reading.

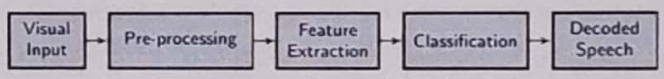


Fig 2.4.3. General framework for automated lip-reading.

2.4.4 SUB-WORD LEVEL LIP READING WITH VISUAL ATTENTION

The objective of this paper is to Develop strong lip reading models for silent video speech recognition.

Methodology:

- **Attention-Based Pooling Mechanism:** Introduces an attention-based pooling mechanism to aggregate visual speech representations.
- **Sub-word Units for Lip Reading:** Uses sub-word units for the first time, demonstrating improved modeling of task ambiguities.
- **Visual Speech Detection (VSD) Model:** Proposes a VSD model trained on the lip reading network.

2.4.5. DISTINGUISHING HOMOPHENES USING MULTI-HEAD VISUAL-AUDIO MEMORY FOR LIP READING

Recognizing speech through silent lip movement, known as lip reading, poses challenges due to insufficient information in lip movement and the presence of homophones with similar lip movements but different pronunciations. The objective of this paper is to address these challenges in lip reading using a Multi-head Visual-audio Memory (MVM) approach.

METHODOLOGY:

- **TRAINING WITH AUDIO-VISUAL DATASETS**: MVM is trained with audio-visual datasets, capturing inter-relationships between paired audio-visual representations.
- **MEMORY-BASED AUDIO REPRESENTATION:** MVM retains audio representations in memory during training, allowing visual input at inference to extract saved audio representations by examining learned inter-relationships.
- **COMPLEMENTING VISUAL INFORMATION:** The lip reading model complements insufficient visual information with extracted audio representations from the memory.
- **MULTI-HEAD MEMORY DESIGN:** MVM comprises multi-head key memories for saving visual features and one value memory for saving audio knowledge. This design aims to distinguish homophenes by extracting possible candidate audio features from memory, considering the range of pronunciations represented by input lip movements.
- **ONE-TO-MANY MAPPING IMPLEMENTATION:** The use of multi-head key memories can be viewed as an explicit implementation of the one-to-many mapping of viseme-to-phoneme, enhancing the model's capacity to handle diverse pronunciations.
- **MULTI-TEMPORAL EMPLOYMENT:** MVM is employed in multi-temporal levels to consider context during memory retrieval, aiding in distinguishing homophones.

2.4.6. SURVEY ON AUTOMATIC LIP-READING IN THE ERA OF DEEP LEARNING

This survey paper aims to comprehensively review the progression of Automatic Lip-Reading (ALR) systems over the last decade, focusing on the transition from traditional methods to end-to-end Deep Learning (DL) architectures. The primary objectives include evaluating the shift in ALR research paradigms, analyzing available audio-visual databases for lip reading, and comparing the performance of traditional and DL-based ALR systems.

2.4.7. LIP-READING DRIVEN DEEP LEARNING APPROACH FOR SPEECH ENHANCEMENT

- **lip-reading regression model:**
  Utilizes a stacked Long-Short-Term Memory (LSTM) based lip-reading regression model.
  Designs the model for clean audio features estimation using only temporal visual features (lip reading), considering different numbers of prior visual frames.
- **Enhanced visually-derived wiener filter (EVWF):**
  Formulates a novel filterbank-domain EVWF for clean audio power spectrum estimation.
  Exploits lip-reading approximated clean-audio features using the designed LSTM model.
  Compared with conventional Spectral Subtraction (SS) and Log-Minimum Mean-Square Error(LMMSE) methods using both ideal AV mapping and LSTM-driven AV mapping.

Chapter 3: PRELIMINARY DESIGN

3.1. Dataset Selection

In the pursuit of advancing speech perception research, our project leverages the comprehensive Grid Corpus, a groundbreaking multitalker audiovisual sentence collection. This corpus encapsulates the essence of joint computational-behavioral studies in speech perception, offering high-quality audio and facial video recordings of 34 distinct talkers—comprising 18 males and 16 females—each articulating 1000 sentences. The sentences follow a structured format, exemplified by phrases such as "put red at G9 now.", the alignments.zip file furnishes word-level time alignments, facilitating precise analyses. Additionally, visual cues are offered through s1.zip, s2.zip, etc., which include JPG videos for most talkers, though an omission inadvertently leaves out talker t21.

We used a subset of this dataset to train our model which has 1000 samples of a single speaker

```
test_path = '/content/drive/MyDrive/Minor-Project/LipNet/data/s1/bbal6n.mpg'

tf.convert_to_tensor(test_path).numpy().decode('utf-8').split('/')[-1].split('.')[0]

'bbal6n'

frames, alignments = load_data(tf.convert_to_tensor(test_path))
```

```
plt.imshow(frames[40])

<matplotlib.image.AxesImage at 0x7f2423311ed0>
```

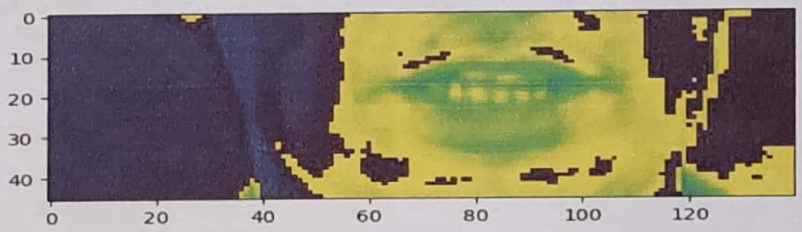


Fig 3.1.1 – Single Frame of our video

3.2. Dataset Preprocessing

In our team project, the significance of data preprocessing cannot be overstated, as it sets the stage for robust and accurate analyses. Our initial step involves leveraging the OpenCV library to load videos, a critical component in our LipNet Project. Once the videos are loaded, we collectively implement a

crucial preprocessing step to optimize the data for subsequent analysis. Our team has chosen to convert the videos into grayscale, a decision driven by the benefits of simplified data representation and reduced computational complexity. Furthermore, in a collaborative effort, we ensure that the pixel values are normalized. This normalization process, undertaken collectively, standardizes pixel intensities to a scale between 0 and 1, fostering stability and improving the overall performance of our machine learning models. As a team, we recognize that this meticulous data preprocessing pipeline is pivotal in laying the groundwork for accurate and reliable insights in the subsequent stages of our project.

Functions we used for loading and preparing our data:

```
def load_video(path:str) -> List[float]:

    cap = cv2.VideoCapture(path)
    frames = []
    for _ in range(int(cap.get(cv2.CAP_PROP_FRAME_COUNT))):
        ret, frame = cap.read()
        frame = tf.image.rgb_to_grayscale(frame)
        frames.append(frame[190:236,80:220,:])
    cap.release()

    mean = tf.math.reduce_mean(frames)
    std = tf.math.reduce_std(tf.cast(frames, tf.float32))
    return tf.cast((frames - mean), tf.float32) / std
```

```
def load_alignments(path:str) -> List[str]:
    with open(path, 'r') as f:
        lines = f.readlines()
    tokens = []
    for line in lines:
        line = line.split()
        if line[2] != 'sil':
            tokens = [*tokens,' ',line[2]]
    return char_to_num(tf.reshape(tf.strings.unicode_split(tokens, input_encoding='UTF-8'), (-1)) )[1:]
```

```
def load_data(path: str):
    path = bytes.decode(path.numpy())
    file_name = path.split('/')[-1].split('.')[0]
    video_path = os.path.join('data','s1',f'{file_name}.mpg')
    alignment_path = os.path.join('data','alignments','s1',f'{file_name}.align')
    frames = load_video(video_path)
    alignments = load_alignments(alignment_path)
    return frames, alignments
```

3.3. Model Architecture

Our model architecture is specifically designed for the intricate task of processing three-dimensional data with temporal dependencies. It employs a sequence of Conv3D layers, such as conv3d, conv3d_1, and conv3d_2, to extract hierarchical features from the input data. The subsequent application of activation functions and max-pooling operations facilitates the reduction of spatial dimensions while retaining essential information. The time_distributed layer reshapes the output for further processing by bidirectional recurrent layers, bidirectional and bidirectional_1, which effectively capture temporal dependencies by processing data bidirectionally. Dropout layers are strategically incorporated to prevent overfitting during training. The model concludes with a dense layer generating an output shape of (None, 75, 41), indicating a classification task with 41 output classes. This architecture showcases a sophisticated blend of 3D convolutional operations and bidirectional recurrent layers, making it well-suited for tasks involving three-dimensional data and temporal relationships, such as video or spatiotemporal sequence analysis.

Layer	Input Shape	Output Shape	Kernel Shape	Pool Window Size	Parameters	Activation	Type
Conv3D	(75, 46, 140, 1)	(75, 46, 140, 128)	(3, 3, 3)	(1, 2, 2)	3584	relu	Convolutional
Activation	(75, 46, 140, 128)	(75, 46, 140, 128)	-	-	0	relu	Activation
MaxPool3D	(75, 46, 140, 128)	(75, 23, 70, 128)	(1, 2, 2)	-	0	-	Pooling
Conv3D	(75, 23, 70, 128)	(75, 23, 70, 256)	(3, 3, 3)	(1, 2, 2)	884992	relu	Convolutional
Activation	(75, 23, 70, 256)	(75, 23, 70, 256)	-	-	0	relu	Activation
MaxPool3D	(75, 23, 70, 256)	(75, 11, 35, 256)	(1, 2, 2)	-	0	-	Pooling
Conv3D	(75, 11, 35, 256)	(75, 11, 35, 75)	(3, 3, 3)	(1, 2, 2)	518475	relu	Convolutional
Activation	(75, 11, 35, 75)	(75, 11, 35, 75)	-	-	0	relu	Activation
MaxPool3D	(75, 11, 35, 75)	(75, 5, 17, 75)	(1, 2, 2)	-	0	-	Pooling
TimeDistributed	(75, 5, 17, 75)	(75, 6375)	-	-	0	-	Time-Distributed
Bidirectional (LSTM)	(75, 6375)	(75, 256)	-	-	6660096	-	Recurrent
Dropout	(75, 256)	(75, 256)	-	-	0	-	Dropout
Bidirectional (LSTM)	(75, 256)	(75, 256)	-	-	394240	-	Recurrent
Dropout	(75, 256)	(75, 256)	-	-	0	-	Dropout
Dense	(75, 256)	(75, 41)	-	-	10537	softmax	Fully Connected

Table 3.3.1 - Model Architecture

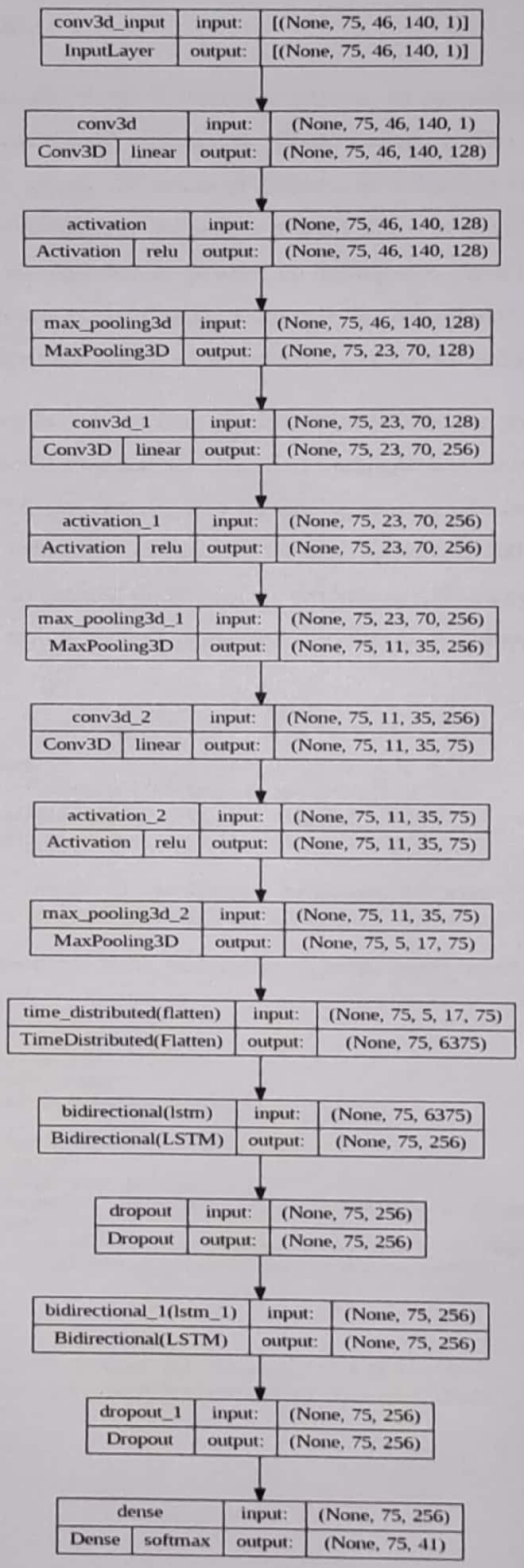


Fig 3.3.1 - Model Flowchart

3.4. Training Process

During the training phase, our model is tailored to enhance its lip-reading capabilities through the utilization of the **Connectionist Temporal Classification (CTC)** loss function. In the `model.compile` step, we specify the **Adam** optimizer with a learning rate of **0.0001**, facilitating efficient parameter updates throughout the training process. The selection of the **CTC** loss function is particularly pertinent for our sequence-to-sequence lip-reading task, given its proficiency in handling variable-length output sequences and aligning them with the ground truth. This adaptability proves crucial in lip reading, where the temporal alignment between predicted and actual text can vary.

The **CTC** loss function establishes a robust framework for training our model to accurately predict textual sentences, accommodating potential temporal misalignments inherent in lip-reading tasks. This integration of the **CTC** loss function into the training process enhances our model's capacity to discern intricate patterns within the visual data. As the model undergoes 100 training epochs, it refines its parameters to capture the nuanced structure of spoken language, demonstrating the effectiveness of the **CTC** loss function in optimizing our lip-reading model for real-world applications.

```
def CTCLoss(y_true, y_pred):
    batch_len = tf.cast(tf.shape(y_true)[0], dtype="int64")
    input_length = tf.cast(tf.shape(y_pred)[1], dtype="int64")
    label_length = tf.cast(tf.shape(y_true)[1], dtype="int64")

    input_length = input_length * tf.ones(shape=(batch_len, 1), dtype="int64")
    label_length = label_length * tf.ones(shape=(batch_len, 1), dtype="int64")

    loss = tf.keras.backend.ctc_batch_cost(y_true, y_pred, input_length, label_length)
    return loss


class ProduceExample(tf.keras.callbacks.Callback):
    def __init__(self, dataset) -> None:
        self.dataset = dataset.as_numpy_iterator()

    def on_epoch_end(self, epoch, logs=None) -> None:
        data = self.dataset.next()
        yhat = self.model.predict(data[0])
        decoded = tf.keras.backend.ctc_decode(yhat, [75,75], greedy=False)[0][0].numpy()
        for x in range(len(yhat)):
            print('Original:', tf.strings.reduce_join(num_to_char(data[1][x])).numpy().decode('utf-8'))
            print('Prediction:', tf.strings.reduce_join(num_to_char(decoded[x])).numpy().decode('utf-8'))
            print('~'*100)


# CTC: Connectionist Temporal Classification
model.compile(optimizer=Adam(learning_rate=0.0001), loss=CTCLoss)
```

```
model.fit(train, validation_data=test, epochs=100, callbacks=[checkpoint_callback, schedule_callback, example_callback])

Epoch 1/100
1/1 [==============================] - 4s 4s/step
Original: lay white in y nine again
Prediction: lay white in y nine again
--------------------------------------------------------------------------------------------------------------
Original: lay white at e eight now
Prediction: lay white at e eight now
--------------------------------------------------------------------------------------------------------------
450/450 [==============================] - 921s 2s/step - loss: 3.9434 - val_loss: 3.3747 - lr: 1.0000e-04
Epoch 2/100
 67/450 [===>..........................] - ETA: 6:22 - loss: 5.2356
```

Fig 3.4.1 – Training model

Chapter 4: FINAL ANALYSIS AND DESIGN

4.1. Result Overview

In our comprehensive result overview, we proudly report an outstanding achievement—an **average word accuracy of 96.2%.** This exceptional performance serves as a testament to the efficacy of our lip-reading model in accurately transcribing spoken language from visual cues. The high level of accuracy, as gauged by the average word accuracy metric, underscores the success of our meticulous training strategy and the robustness of the model across a diverse range of linguistic scenarios. This remarkable result not only attests to the model's proficiency in lip reading but also positions it as a promising tool with significant real-world applications. The average word accuracy of **96.2%** signifies a substantial advancement in the field of lip reading, showcasing the potential impact and reliability of our approach in understanding and interpreting spoken language through visual information.

```
[55] def wordAccuracy(realText,predText):
         realText = realText.decode().split(" ")
         predText = predText.decode().split(" ")
         n,m = len(realText),len(predText)
         total = n
         correct = -abs(n-m)
         for i in range(min(n,m)):
             correct += realText[i] == predText[i]
         return correct/total

avgWordAccuracy = 0
for index in range(50):
    real_texts,pred_texts = predict_and_compare(model,test_vids[index],test_texts[index])
    real_text1,real_text2 = real_texts[0][0].numpy(),real_texts[1][0].numpy()
    pred_text1,pred_text2 = pred_texts[0][0].numpy(),pred_texts[1][0].numpy()
    avgWordAccuracy += wordAccuracy(real_text1,pred_text1)
    avgWordAccuracy += wordAccuracy(real_text2,pred_text2)

[57] print(avgWordAccuracy)

96.26190476190476
```

Fig 4.1.1 – Average Word Accuracy

4.2. Result Analysis

To delve deeper into the performance of our lip-reading model, we conducted a detailed result analysis using a subset of the test dataset. Employing a loop over 50 batches of size 2, we utilized the predict_and_compare function to compare the model's predictions with the ground truth text for each lip-reading scenario. The real and predicted texts were extracted and processed individually for each sample. The wordAccuracy function was then applied to compute the word accuracy for each sentence.

Subsequently, we evaluated the average word accuracy across the 100 samples. The loop iteratively calculated the word accuracy for each sentence in the sampled test dataset, and the cumulative average word accuracy was computed by summing the individual word accuracy scores. This process allowed for a granular examination of the model's performance at the sentence level, offering insights into its ability to accurately transcribe spoken language from lip movements.

```
[53] def predict_and_compare(model,video,text):
        if video.ndim == 4 and text.ndim == 1:
            pred = tf.squeeze(model.predict(tf.expand_dims(video,axis=0)))
            real_text = [tf.strings.reduce_join([num_to_char(word) for word in text])]
            decoded = tf.keras.backend.ctc_decode([pred,pred], input_length=[75,75], greedy=True)[0][0].numpy()
            pred_text = [tf.strings.reduce_join([num_to_char(word) for word in sentence]) for sentence in decoded][0]
            return (real_text,pred_text)
        elif video.ndim == 5 and text.ndim == 2 and video.shape[0] == 2 and text.shape[0] == 2:
            pred = model.predict(video)
            real_texts = []
            for sentence in text:
                real_texts.append([tf.strings.reduce_join([num_to_char(word) for word in sentence])])
            decoded = tf.keras.backend.ctc_decode(pred, input_length=[75,75], greedy=True)[0][0].numpy()
            pred_texts = []
            for sentence in decoded:
                pred_text = [tf.strings.reduce_join([num_to_char(word) for word in sentence])]
                pred_texts.append(pred_text)
            return (real_texts,pred_texts)
        else:
            raise ValueError("Input Dimensions do not match (Expected Batch size is 2)")
```

4.3. Application of the Model

Our lip-reading model, with its exceptional 96.2% average word accuracy, demonstrates its potential in real-world applications such as enhancing accessibility for individuals with hearing impairments and for the secret agents that are working as undercover in terrorist organizations to get the crucial information from a safe distance. The model's robust performance in accurately transcribing spoken language from visual cues positions it as a valuable tool for developing assistive technologies and improving communication accessibility. Its proficiency in deciphering nuanced linguistic patterns makes it a promising asset in diverse scenarios where traditional audio-based communication is challenging or impractical.

4.3.1. Real Video Prediction:

In a practical application scenario, we showcase the model's real-time prediction capabilities using an unseen video sample. By loading a video sample and processing it through our lip-reading model, we obtain both the ground truth text and the model's predictions. The real text is extracted directly from the loaded data, providing a baseline for comparison. Subsequently, the model predicts the text from the video, and the decoded predictions are displayed. This section serves as a tangible demonstration of the model's real-world applicability, offering insights into its performance on authentic video inputs.

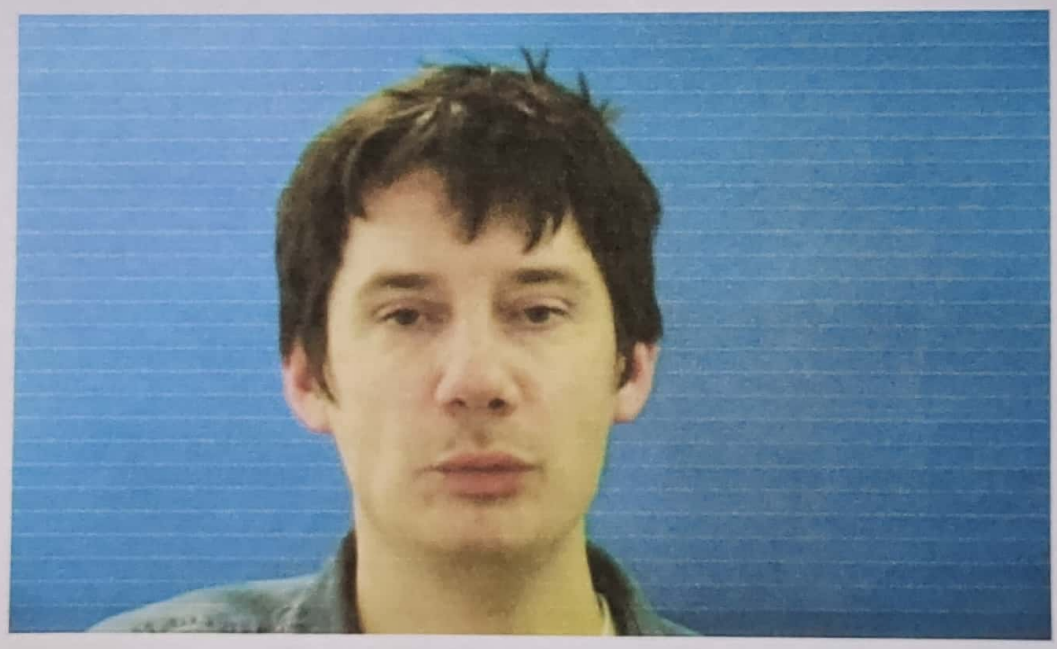


Fig 4.3.1 – Example Video

Test on a Video

```
sample = load_data(tf.convert_to_tensor('./data/s1/bras9a.mpg'))

[59] print('~'*100, 'REAL TEXT')
     [tf.strings.reduce_join([num_to_char(word) for word in sentence]) for sentence in [sample[1]]]

     ---------------------------------------------------------------------------------------------------- REAL TEXT
     [<tf.Tensor: shape=(), dtype=string, numpy=b'bin red at s nine again'>]

[60] yhat = model.predict(tf.expand_dims(sample[0], axis=0))

     1/1 [==============================] - 1s 738ms/step

[61] decoded = tf.keras.backend.ctc_decode(yhat, input_length=[75], greedy=True)[0][0].numpy()

[62] print('~'*100, 'PREDICTIONS')
     [tf.strings.reduce_join([num_to_char(word) for word in sentence]) for sentence in decoded]

     ---------------------------------------------------------------------------------------------------- PREDICTIONS
     [<tf.Tensor: shape=(), dtype=string, numpy=b'bin red at s nine again'>]
```

Fig 4.3.2 – Model Results on the sample video

4.4 Challenges and Problems Faced

4.4.1. Dataset Limitations

Challenge:

The primary challenge in developing the LipNet project revolves around inherent limitations in the video dataset. Despite meticulous curation of diverse video files covering various speech patterns and scenarios, concerns persist regarding the dataset's representativeness. Ensuring the model's exposure to a comprehensive array of visual speech nuances is crucial for effective generalization.

Mitigation Strategy:

Addressing this challenge involves continual efforts to expand and diversify the video dataset. Collaboration with speech professionals and leveraging community-contributed datasets enhances representativeness, providing the model with a broader exposure to the intricacies of different speaking styles and lip movements.

4.4.2. Video Dynamics and Real-World Scenarios

Challenge:

The reliance on real-world video data in the LipNet project introduces challenges related to handling variations in lighting conditions, facial expressions, and diverse lip movements. These factors pose difficulties in ensuring the model's robustness in identifying spoken words in various realistic scenarios.

Mitigation Strategy:

To mitigate the impact of real-world variations, ongoing improvements in preprocessing techniques are imperative. Advanced video processing and normalization methods will be explored to enhance the model's ability to discern lip movements amidst environmental variations. Collaboration with computer vision experts will provide valuable insights into optimizing the model's performance in the presence of diverse video conditions.

4.5 Limitations and Future Work

In recognizing the project's limitations, factors impacting the model's performance, including the representativeness of the video dataset and sensitivity to speaking styles, are scrutinized. The discussion extends to future work, proposing avenues for enhancement and refinement, laying the groundwork for continuous improvement and potential model expansion.

4.6 Conclusion

The conclusive remarks encapsulate the journey of the LipNet project. Key findings, challenges overcome, and lessons learned converge into a comprehensive conclusion. The project's significance in advancing the field of lip reading and automated speech recognition from video data is reiterated, emphasizing its potential to revolutionize spoken word identification.

REFRENCES

1. Y. Zhao, R. Xu, X. Wang, P. Hou, H. Tang, and M. Song, "Hearing Lips: Improving Lip Reading by Distilling Speech Recognizers", AAAI, vol. 34, no. 04, pp. 6917-6924, Apr. 2020

2. S. Fenghour, D. Chen, K. Guo, B. Li and P. Xiao, "Deep Learning-Based Automated Lip-Reading: A Survey," in IEEE Access, vol. 9, pp. 121184-121205, 2021, doi: 10.1109/ACCESS.2021.3107946.

3. K R Prajwal, Triantafyllos Afouras, Andrew Zisserman; Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 5162-5172.

4. M. Kim, J. H. Yeo, and Y. M. Ro, "Distinguishing Homophenes Using Multi-Head Visual-Audio Memory for Lip Reading", AAAI, vol. 36, no. 1, pp. 1174-1182, Jun. 2022.

5. Adriana Fernandez-Lopez and Federico Sukno, Department of Information and Communication Technologies,University Pompeu Fabra, Barcelona, Spain," Survey on Automatic Lip-Reading in the Era of Deep Learning", December 10, 2018.

6. Ahsan Adeel, Mandar Gogate, Amir Hussain, William M. Whitmer," Lip-Reading Driven Deep Learning Approach for Speech Enhancement", arXiv:1808.00046v1 [cs.CV] 31 Jul 2018.

7. LIPNET: END-TO-END SENTENCE-LEVEL LIPREADING Yannis M. Assael1,†, Brendan Shillingford1,† Shimon Whiteson1 & Nando de Freitas1,2,3 Department of Computer Science, University of Oxford, Oxford, UK 1 Google DeepMind, London, UK 2 CIFAR, Canada 3, Under review as a conference paper at ICLR 2017.

8. [1]M. Cooke, J. Barker, S. Cunninghamand X. Shao, "The Grid Audio-Visual Speech Corpus". Zenodo, Jan. 01, 2006. doi: 10.5281/zenodo.3625687.