

Networks for Automated Lip Reading

CHAPTER 1

INTRODUCTION

Lip reading, or visual speech recognition, involves interpreting human speech by visually observing the movements of the lips, facial expressions, and sometimes other visual cues. Creating a deep learning model that can accurately lip read can have significant applications in various fields such as communication, accessibility, security, and human-computer interaction.

1.1 Problem statement

Traditional lip reading models face several challenges like models often struggle to accurately interpret lip movements due to the inherent ambiguity, Human lips come in various shapes, sizes, and expressions, influenced by factors such as language, accent, and individual differences models may fail to generalize across this variability, models can be significantly affected by environmental factors such as lighting conditions, camera angles, and background clutter.

Our model utilizes Deep Learning Architectures to improve the accuracy of lip reading models, Augment existing lip reading datasets with variations in lip shapes, expressions, and environmental conditions to enhance model robustness.

Therefore, there is a strong need for an automated lip reading system that can provide an alternative means of understanding speech in noisy environments or assist those with hearing disabilities.

- Collecting a dataset of videos with corresponding text transcripts.
- Preprocessing the videos to extract frames and convert them into suitable representations, such as spectrograms.
- Designing a deep learning architecture, such as a convolutional neural network (CNN) or recurrent neural network (RNN), to learn the mapping between visual input (lip movements) and text output (spoken words).
- Training the model using TensorFlow, optimizing its parameters, and tuning hyperparameters.
- Evaluating the model's performance on a separate test set, measuring metrics like accuracy, word error rate, or other relevant evaluation criteria.
- Iterating on the model design and training process to improve performance as needed.

1.2 Objectives and Scope of Project

OBJECTIVES:

1. Develop an Accurate Lip Reading Model:

- Create a deep learning model to accurately transcribe spoken words from silent video footage of lip movements.

2. Achieve Robust Performance:

Networks for Automated Lip Reading

- Ensure the model generalizes well across different speakers, accents, and varying video conditions.

3. Utilize Advanced Deep Learning Techniques:

- Implement state-of-the-art neural network architectures (e.g., CNNs, RNNs) and explore advanced methods (e.g., 3D CNNs, attention mechanisms).

SCOPE OF PROJECT:

The scope of this research encompasses several key areas in the application of deep learning and neural networks of Revolutionizing Visual Speech Recognition: Harnessing Neural Networks for Automated Lip Reading.

1. Literature Review and Background:

- Survey of Existing Techniques: Comprehensive review of current methodologies in visual speech recognition and lip reading, including traditional approaches and modern deep learning methods.
- Technological Foundations: Overview of deep learning principles, neural network architectures, and their applications in computer vision and speech recognition.

2. Data Collection and Preprocessing:

- Dataset Acquisition: Collection of video datasets containing diverse speakers and speech scenarios (e.g., Lip Reading in the Wild (LRW) dataset).
- Preprocessing Techniques: Methods for preprocessing video data, including frame extraction, face and lip region detection, normalization, and augmentation to enhance model training.

3. Model Architecture and Development:

- Neural Network Design: Exploration and design of neural network architectures suitable for lip reading, such as Convolutional Neural Networks (CNNs) for feature extraction and Recurrent Neural Networks (RNNs) or Transformers for sequence modeling.
- Integration of DeepSpeech2: Implementation of the DeepSpeech2 architecture to leverage its advanced speech recognition capabilities in the context of visual data.

4. Training and Optimization:

- Model Training: Procedures for training the neural network models using the prepared datasets, including splitting data into training, validation, and test sets.

Networks for Automated Lip Reading

- Hyperparameter Tuning: Optimization of model hyperparameters to improve performance, using techniques such as grid search or random search.
- Regularization Techniques: Implementation of methods to prevent overfitting, such as dropout, data augmentation, and early stopping.

5. Evaluation and Validation:

- Performance Metrics: Definition and calculation of key performance metrics, such as accuracy, Word Error Rate (WER), and other relevant measures for assessing model effectiveness.
- Model Validation: Validation of the trained models on separate validation and test datasets to ensure generalization and robustness.

6. Deployment and Application:

- Real-time Implementation: Development of a system or application that demonstrates the lip reading model's capabilities in real-time or on pre-recorded video content.
- User Interface: Design of a user-friendly interface to interact with the lip reading system, enabling practical use cases.

7. Documentation and Reporting:

- Comprehensive Documentation: Detailed documentation of the entire research process, including code, methodologies, and experimental results.
- Research Paper: Compilation of findings into a formal research paper, outlining the significance, methods, results, and future work in the field of visual speech recognition.

1.3 Motivation of project

1. Advancing Accessibility:

- Enhancing Communication: Providing an effective communication tool for individuals with hearing impairments.

- Bridging Gaps: Enabling more seamless interaction in noisy environments where traditional audio-based speech recognition fails.

2. Technological Innovation:

Networks for Automated Lip Reading

-State-of-the-Art Techniques: Utilizing cutting-edge neural network architectures like DeepSpeech2 and other advanced models.

-Interdisciplinary Approach: Combining computer vision, natural language processing, and deep learning for innovative solutions.

3. Improving Accuracy:

- Enhanced Precision: Improving the accuracy of speech recognition systems through visual cues, complementing traditional audio-based methods.
- Reducing Errors: Minimizing misinterpretations caused by accents, dialects, and speech impediments by leveraging lip movements.

4. Real-World Applications:

- Security and Surveillance: Enhancing security systems by enabling silent communication and monitoring in sensitive areas.
- Healthcare: Assisting in the diagnosis and monitoring of speech-related disorders.

5. User Experience:

-Interactive Systems: Enabling more natural and intuitive interactions with machines and digital assistants.

-Hands-Free Operation: Facilitating hands-free control in various applications, from consumer electronics to industrial machinery.

6. Research and Development:

-Benchmarking Progress: Establishing new benchmarks and datasets for lip reading, driving further research and development.

-Cross-Disciplinary Impact: Inspiring innovations across related fields such as linguistics, psychology, and cognitive science.

7. Commercial Potential:

-Market Opportunities: Tapping into new market opportunities in sectors like telecommunications, gaming, and customer service.

Networks for Automated Lip Reading

-Competitive Edge: Offering businesses a competitive edge with advanced human-computer interaction technologies.

By focusing on these key areas, this research aims to revolutionize visual speech recognition, making significant contributions to both the scientific community and real-world applications.

CHAPTER 2 LITERATURE SURVEY

Authors/Year of Publication	Title of the Article	Methods Used	Results	Remarks
End-to-End Lip Reading with Transformer Networks, Ma, X., Liu, Y., Wang, S., Wu, W. Proceedings of the European Conference on Computer Vision (ECCV), 2022	End-to-End Lip Reading with Transformer Networks	Transformer-based approach outperforms traditional CNN-RNN models on several lip reading datasets, demonstrating the potential of transformers for this task.	The authors propose a transformer-based model for lip reading, leveraging the attention mechanism to handle long-range dependencies in video sequences.	Presents a significant advancement in the field of lip reading by introducing a novel transformer-based model. By leveraging the attention mechanism inherent in transformer networks, the authors effectively address the challenge of handling long-range dependencies in video sequences, which is crucial for accurate lip reading.

Revolutionizing Visual Speech Recognition: Harnessing Neural

Networks for Automated Lip Reading

Souheil Fenghour, Daqing Chen in “Lip Reading Sentences Using Deep Learning with Only Visual Cues” published in November 9, 2020, accepted. November	Lip Reading Sentences Using Deep Learning with Only Visual Cues	Deep Learning, focusing on visual cues.	Involves the development of a deep learning model specifically designed for lip-reading using only visual information.	This paper seems to emphasize the utilization of visual cues alone for lip-reading tasks, which could be beneficial for scenarios where audio information is not available or
18, 2020, date of publication November 26, 2020, date of current version December 11, 2020.				reliable.
Ahsan Adeel, Mandar Gogate, Amir Hussain, and William M. Whitmer in “Lip-Reading Driven Deep Learning Approach for Speech Enhancement” IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTATIONAL INTELLIGENCE.	Lip-Reading Driven Deep Learning Approach for Speech Enhancement	Deep Learning, likely with a focus on lip-reading	involves the development of a deep learning approach for enhancing speech using lip-reading cues.	This paper seems to explore how lip-reading can be used to improve speech enhancement, possibly by incorporating visual information into the enhancement process to address noisy or degraded audio signals.

Networks for Automated Lip Reading

H. Kulkarni and D. Kirange, "Artificial Intelligence: A Survey on Lip-Reading Techniques,"2019 10th International Conference on Computing, Communication and Networking Technologies	Artificial Intelligence: A Survey on Lip-Reading Techniques	Survey methodology, likely involving literature review and analysis.	Involves summarizing and analyzing various lip-reading techniques discussed in the surveyed literature.	This paper provides a comprehensive overview of existing lip-reading techniques, highlighting trends, challenges, and future directions in the field. It serves as a valuable resource for understanding the landscape of lip-reading research.
--	---	--	---	---

(ICCCNT),2019				
T. Thein and K. M. San, "Lip movements recognition towards an automatic lip-reading system for Myanmar consonants," 2018 12th International Conference on Research Challenges in Information Science(RCIS),2018, pp.1-6, Doi:10.1109/RCIS.2018.8406660.	Lip movements recognition towards an automatic lip-reading system for Myanmar consonants	Lip movement recognition.	Successful recognition of lip movements for Myanmar consonants	Developing automatic lip-reading systems tailored to specific languages, such as Myanmar, is essential for improving accessibility and communication for speakers of those languages.

Networks for Automated Lip Reading

W. Nittaya, K. Wetchasit and K. Silanon, "Thai Lip-Reading CAI for Hearing Impairment Student," 2018 Seventh ICT International Student Project Conference (ICTISPC), 2018	Thai Lip-Reading CAI for Hearing Impairment Student	Development of a Thai lip-reading computer-assisted instruction (CAI) system.	Possibly improved lip-reading skills for hearing-impaired students.	Creating specialized lip-reading tools for specific user groups, such as hearing-impaired students, can significantly enhance their communication abilities and educational experiences.
---	---	---	---	--

CHAPTER 3

PROBLEM ANALYSIS & SYSTEM DESIGN

3.1 Existing system

1. Data Collection and Preprocessing:

Collects videos of individuals speaking for lip movement analysis.

Applies basic preprocessing techniques like facial detection and lip segmentation.

2. Neural Network Architecture:

Utilizes convolutional and recurrent neural networks for feature extraction and modeling.

Trained using supervised learning on labeled datasets of lip movements.

3. Evaluation and Integration:

Evaluates performance on test datasets for accuracy assessment.

Integrates into applications like accessibility tools and security systems for real-world usage.

3.2 Proposed system

Deep Learning for Visual Speech Recognition System

1.Feature Extraction with CNN:

-Convolutional Neural Networks (CNNs): Use a CNN to extract spatial features from the video frames. CNNs are effective at capturing spatial hierarchies and details within the image, which are crucial for distinguishing lip movements.

-CNN Architecture: Implement a standard CNN architecture (e.g., VGG, ResNet) tailored for lip reading, which could include several convolutional layers followed by pooling layers to reduce dimensionality while preserving essential features.

2.Sequence Modeling with RNN:

-Recurrent Neural Networks (RNNs): Pass the sequence of feature vectors obtained from the CNN through an RNN to capture temporal dependencies between consecutive frames.

-Long Short-Term Memory (LSTM) / Gated Recurrent Units (GRU): Use LSTM or GRU layers to handle long-range dependencies and avoid vanishing gradient issues typically associated with vanilla RNNs.

3.Integration with DeepSpeech2:

- DeepSpeech2 Architecture: Combine the strengths of the DeepSpeech2 model, which integrates CNN and RNN components for speech recognition.

-Adaptation for Visual Input: Modify the DeepSpeech2 architecture to handle visual input (lip movements) instead of audio signals, ensuring the model can process and recognize visual speech patterns.

-The Deep Neural Network (DNN) part produces a probability distribution $P_t(c)$ over vocabulary characters c per each time step t .

4.Training and Optimization:

-Dataset: Utilize a comprehensive dataset such as Lip Reading in the Wild (LRW) to train the model, ensuring it is exposed to diverse lip movements and speaking styles.

-Loss Function: Implement a suitable loss function (e.g., Connectionist Temporal Classification (CTC) loss) to handle the sequence-to-sequence nature of lip reading.

Networks for Automated Lip Reading

-Hyperparameter Tuning: Optimize model hyperparameters (e.g., learning rate, batch size, number of layers) to achieve the best performance.

5.Performance Evaluation:

-Metrics: Evaluate the model using metrics such as accuracy, Word Error Rate (WER), and frame-level accuracy.

-Validation and Testing: Conduct thorough validation and testing using separate datasets to ensure the model generalizes well to unseen data.

This system design leverages the strengths of CNNs for feature extraction, RNNs for sequence modeling, and the DeepSpeech2 architecture for integrating these components into a robust visual speech recognition system. This approach aims to deliver high accuracy in lip reading by effectively capturing both spatial and temporal information from video inputs.

3.3System Architecture

Convolutional neural network architecture

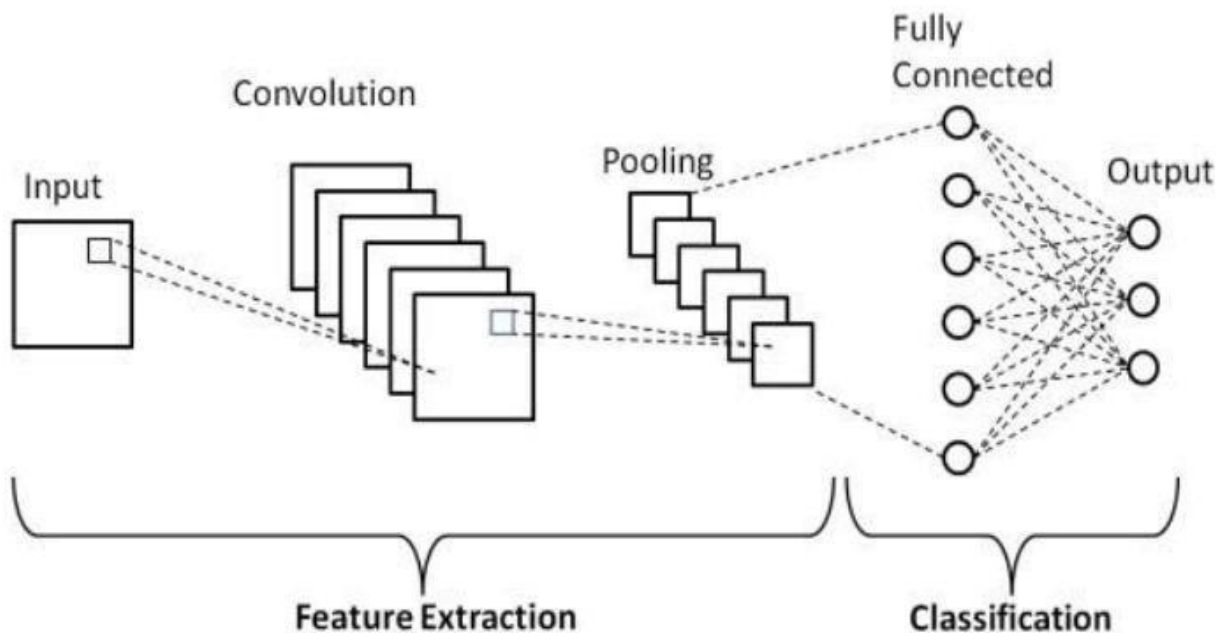


Fig.4.3.1 Block Diagram of CNN Architecture

Networks for Automated Lip Reading

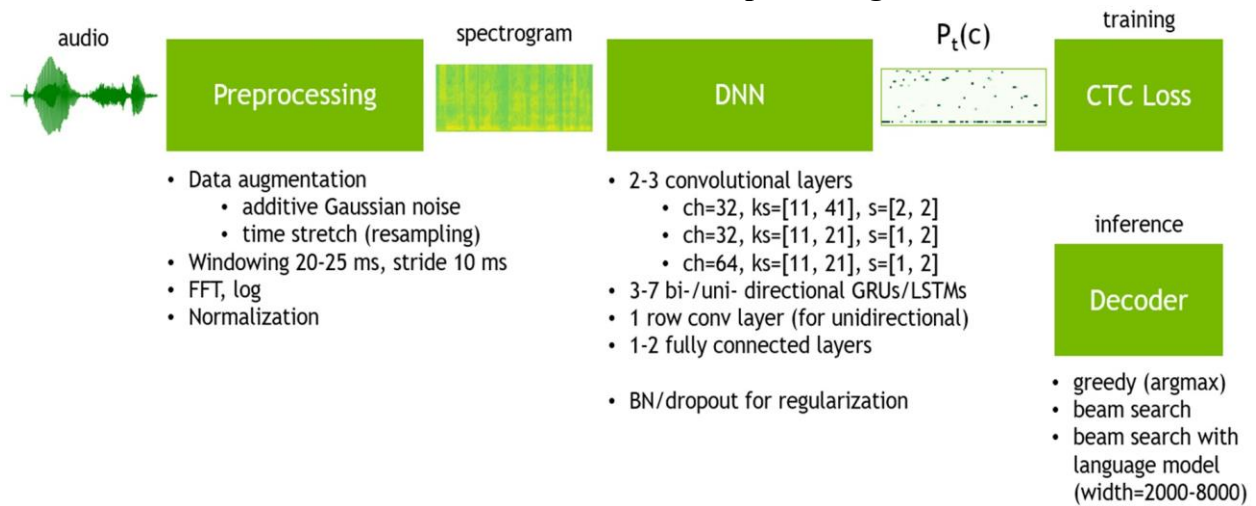


Fig 4.3.2. Block Diagram of DNN Architecture

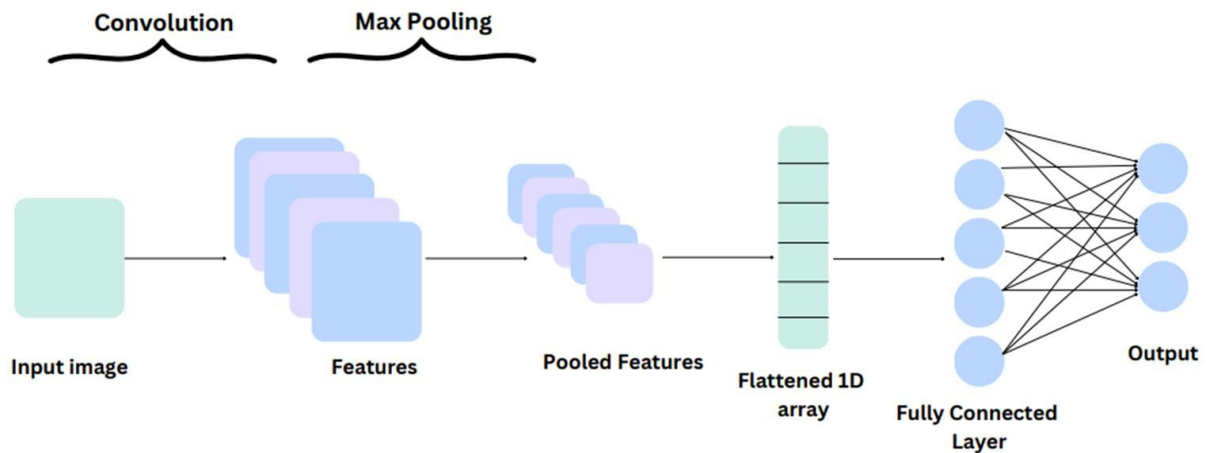


Fig 4.3.3 Working of CNN

Networks for Automated Lip Reading

The Convolution Operation

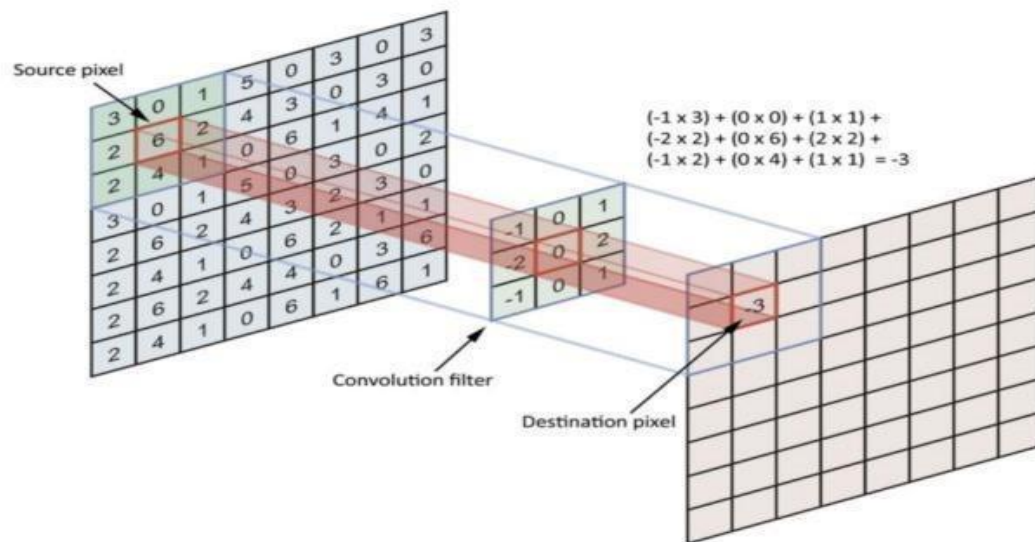


Fig 4.3.4 Convolution Operation

- MTCNN algorithm: The MTCNN (Multi-Task Cascaded Convolutional Networks) algorithm is a deep learning-based face detection and alignment method that uses a cascading series of convolutional neural networks (CNNs) to detect and localize faces in digital images or videos.
- CNN uses the sliding window approach to extract the features.
- It provides features like padding, pooling layers which makes it more easier and accurate in detecting the image.
- Word Error Rate (WER) is the main evaluation metric. In order to get words out of a trained model one needs to use a decoder. Decoder converts a probability distribution over characters into text.
- A beam search decoder with language model re-scoring allows checking many possible decodings (beams) at once with assigning a higher score for more probable N-grams according to a given language model. The language model helps to correct misspelling errors. The downside is that it is significantly slower than a greedy decoder.
 - o Software and Hardware requirements

Networks for Automated Lip Reading

3.4 Software Requirements

Operating System

Windows 10 or later

macOS Catalina or later

Linux (Ubuntu 18.04 or later recommended)

Programming Languages and Libraries

Python: Version 3.6 or later

TensorFlow: TensorFlow 1.0+ ,TensorFlow is an open-source deep learning framework that provides a comprehensive ecosystem for building and deploying machine learning models.

OpenCV: This package provides OpenCV, a library for computer vision tasks such as reading, processing, and displaying images and videos.

NumPy: For numerical operations

Matplotlib/Seaborn: For data visualization

Keras: Keras 2.0+ ,For using the high-level API for TensorFlow

Gdown: gdown is a tool for downloading files from Google Drive, which can be useful for acquiring datasets or pre-trained models stored on Google Drive.

imageio: Imageio is a library for reading and writing image data, supporting a wide range of image file formats

PIP: For package installation

Development Tools

Jupyter Notebook: For interactive development and experimentation

Anaconda: For managing Python environments and dependencies

Integrated Development Environment (IDE): Such as PyCharm, VS Code, or JupyterLab

Version Control

Git: For version control

GitHub/GitLab/Bitbucket: For code repository hosting

Dependencies Management

pip: Python package installer for managing libraries

virtualenv or conda: For creating isolated environments

Additional Tools ffmpeg: For video processing and conversion

Networks for Automated Lip Reading

Hardware Requirements

Processor (CPU)

Minimum: Quad-core processor (e.g., Intel Core i5 or AMD Ryzen 5)

Recommended: High-performance multi-core processor (e.g., Intel Core i7/i9 or AMD Ryzen7/9)

Memory (RAM)

Minimum: 16 GB

Recommended: 32 GB or more, especially for handling large datasets and complex models

Graphics Processing Unit (GPU)

Minimum: NVIDIA GPU with CUDA support (e.g., GTX 1060)

Recommended: NVIDIA GPU with at least 8 GB VRAM (e.g., RTX 2070/2080, RTX 30 series)

CUDA: Install NVIDIA CUDA Toolkit and cuDNN for GPU acceleration

Storage

Minimum: 256 GB SSD

Recommended: 512 GB SSD or more for faster data access and storage of large datasets

Display

A monitor with a resolution of at least 1080p for better visualization and coding experience

Peripherals

Webcam: For testing real-time lip reading applications

Microphone: For synchronous audio-visual data collection (optional)

Cloud Services (Optional)

Cloud Computing Services

Google Cloud Platform (GCP): TensorFlow integration, AI Platform

Amazon Web Services (AWS): EC2 instances with GPU, SageMaker

Microsoft Azure: Virtual Machines with GPU, Azure Machine Learning

Storage Services

Google Cloud Storage: For storing large datasets and model checkpoints

Amazon S3: Scalable storage solution

Azure Blob Storage: For efficient data storage and access

Networks for Automated Lip Reading

Colab Notebooks

Google Colab: Free GPU/TPU for development and experimentation

Networking and Infrastructure

Internet Connection

Stable high-speed internet for downloading datasets, software dependencies, and accessing cloud services

Local Network

For distributed training setups, if multiple machines or GPUs are used.

CHAPTER 4

IMPLEMENTATION

4.1 Overview of system implementation

Implementing a system for Revolutionizing Visual Speech Recognition (VSR) using neural networks involves several key components and steps. Here's an overview of how such a system can be implemented:

1. Data Acquisition and Preprocessing
 - Dataset Collection: Gather large-scale video datasets suitable for training VSR models, such as GRID, LRW, or custom datasets.
 - Preprocessing:
 - Face Detection and Tracking: Use computer vision techniques to detect and track faces in video frames.
 - Lip Region Extraction: Extract the region of interest (ROI) containing the lips from each frame for focused analysis.
 - Normalization: Preprocess images to ensure consistent lighting, scale, and alignment across frames.
2. Model Selection and Architecture Design
 - Neural Network Architecture: Choose suitable architectures based on recent advancements in VSR, such as:
 - Convolutional Neural Networks (CNNs): For spatial feature extraction from lip images.
 - Recurrent Neural Networks (RNNs) or Transformers: For capturing temporal dependencies and long-range dependencies in lip movements.
 - Multimodal Integration: Incorporate audio features alongside visual cues to enhance model robustness and accuracy.
3. Training and Evaluation

Networks for Automated Lip Reading

- Data Splitting: Divide the dataset into training, validation, and test sets.
- Model Training:
 - Configure the model with appropriate loss functions (e.g., categorical cross-entropy for classification tasks) and optimizers (e.g., Adam or RMSprop).
 - Train the model on the training dataset, monitoring metrics such as loss and accuracy.
- Validation: Evaluate the model's performance on the validation set to tune hyperparameters and prevent overfitting.
- Testing: Assess the final model's performance on the test set using metrics like Word Error Rate (WER) or accuracy.
- 4. Real-Time Processing and Deployment
 - Optimization: Optimize the trained model for deployment on various platforms, including edge devices or cloud servers.
 - Real-Time Capabilities: Implement techniques (e.g., model pruning, quantization) to ensure real-time inference for applications requiring low latency.
 - Deployment: Integrate the model into the target application environment, ensuring compatibility with hardware and software requirements.
- 5. Ethical and Regulatory Considerations
 - Privacy and Security: Implement measures to protect user privacy when handling facial and speech data.
 - Bias Mitigation: Validate and mitigate biases that may arise from training data or model predictions, ensuring fairness across diverse user groups.
 - Compliance: Adhere to regulatory guidelines and standards relevant to data privacy and AI deployment in specific domains (e.g., healthcare, security).
- 6. Continuous Improvement and Maintenance
 - Model Updates: Periodically retrain the model with new data to adapt to evolving speech patterns and environmental conditions.
 - Feedback Loop: Incorporate user feedback to refine the system's performance and usability over time.
 - Monitoring: Establish monitoring mechanisms to detect and address issues related to model performance degradation or system errors.

4.2 Modules description

In the context of implementing a system for Revolutionizing Visual Speech Recognition (VSR) using neural networks, here are the key modules and their descriptions:

Networks for Automated Lip Reading

1. Data Acquisition and Preprocessing

Description: This module focuses on acquiring suitable video datasets and preparing them for training neural networks for VSR.

- **Data Collection:** Gather large-scale video datasets such as GRID, LRW, or custom datasets containing diverse speakers and speech contexts.
- **Face Detection and Tracking:** Utilize computer vision techniques (e.g., Haar cascades, deep learning-based detectors) to detect and track faces in video frames.
- **Lip Region Extraction:** Extract the region of interest (ROI) containing the lips from each frame to isolate visual cues relevant to speech.
- **Normalization:** Preprocess images to standardize lighting, scale, and alignment across frames to reduce variability in training data.

2. Model Selection and Architecture Design

Description: This module involves selecting appropriate neural network architectures and designing them to effectively learn from visual speech data.

- **Neural Network Architecture:** Choose architectures suitable for VSR, such as:
 - **Convolutional Neural Networks (CNNs):** For extracting spatial features from lip images.
 - **Recurrent Neural Networks (RNNs) or Transformers:** To capture temporal dependencies and long-range dependencies in lip movements.
- **Multimodal Integration:** Incorporate audio features alongside visual data to enhance model robustness and accuracy in recognizing spoken words.

3. Training and Evaluation

Description: This module covers the training process of the selected neural network architectures and evaluating their performance.

- **Data Splitting:** Divide the dataset into training, validation, and test sets to train and assess model performance.
- **Loss Functions and Optimizers:** Define appropriate loss functions (e.g., categorical cross-entropy) and optimizers (e.g., Adam, RMSprop) for training the model.
- **Training:** Train the model on the training dataset, monitoring metrics like loss and accuracy to optimize model parameters.
- **Validation:** Evaluate the model on the validation set to fine-tune hyperparameters and prevent overfitting.

Networks for Automated Lip Reading

- Testing: Assess the final model's performance on the test set using metrics such as Word Error Rate (WER) or accuracy to validate its generalization ability.

4. Real-Time Processing and Deployment

Description: This module focuses on optimizing and deploying the trained model for real-time inference in practical applications.

- Model Optimization: Optimize the trained model for deployment on various platforms (e.g., edge devices, cloud servers) to ensure efficient inference.
- Real-Time Capabilities: Implement techniques (e.g., model pruning, quantization) to achieve low-latency inference suitable for real-time applications.
- Integration: Integrate the model into the target application environment, ensuring compatibility with hardware and software requirements for seamless deployment.

5. Ethical and Regulatory Considerations

Description: This module addresses ethical and regulatory aspects related to deploying VSR systems, ensuring responsible and compliant usage.

- Privacy and Security: Implement measures to protect user privacy when handling facial and speech data, adhering to data protection regulations.
- Bias Mitigation: Validate and mitigate biases in training data or model predictions to ensure fairness and inclusivity across diverse user groups.
- Compliance: Ensure compliance with regulatory guidelines and standards relevant to data privacy, AI deployment, and specific application domains (e.g., healthcare, security).

6. Continuous Improvement and Maintenance

Description: This module focuses on ongoing maintenance and improvement of the VSR system post-deployment.

- Model Updates: Periodically retrain the model with new data to adapt to evolving speech patterns and environmental conditions, ensuring continued accuracy and performance.
- Feedback Loop: Incorporate user feedback to refine the system's performance and usability, addressing user needs and enhancing overall user experience.
- Monitoring: Establish monitoring mechanisms to detect and address issues related to model performance degradation or system errors, maintaining system reliability and effectiveness.

Networks for Automated Lip Reading

4.3 Code snippets

```
In [ ]: frames, alignments = data.as_numpy_iterator().next()

In [ ]: len(frames)

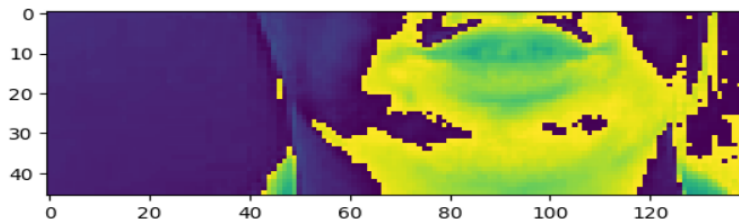
In [ ]: sample = data.as_numpy_iterator()

In [ ]: val = sample.next(); val[0]

In [ ]: imageio.mimsave('./animation.gif', val[0][0], fps=10)

In [34]: # 0: videos, 0: 1st video out of the batch, 0: return the first frame in the video
plt.imshow(val[0][0][35])
```

Out[34]: <matplotlib.image.AxesImage at 0x10c2ead8d30>



4. Setup Training Options and Train

```
In [45]: def scheduler(epoch, lr):
        if epoch < 30:
            return lr
        else:
            return lr * tf.math.exp(-0.1)

In [46]: 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

In [47]: 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)
```

Networks for Automated Lip Reading

```
In [57]: yhat = model.predict(sample[0])

1/1 [=====] - 1s 973ms/step

In [58]: print('~'*100, 'REAL TEXT')
[tf.strings.reduce_join([num_to_char(word) for word in sentence]) for sentence in sample[1]]

~~~~~ REAL TEXT

Out[58]: [<tf.Tensor: shape=(), dtype=string, numpy=b'place white at x six please'>,
<tf.Tensor: shape=(), dtype=string, numpy=b'lay blue in x four now'>]

In [59]: decoded = tf.keras.backend.ctc_decode(yhat, input_length=[75,75], greedy=True)[0][0].numpy()

In [60]: print('~'*100, 'PREDICTIONS')
[tf.strings.reduce_join([num_to_char(word) for word in sentence]) for sentence in decoded]

~~~~~ PREDICTIONS

Out[60]: [<tf.Tensor: shape=(), dtype=string, numpy=b'place white at x six please'>,
<tf.Tensor: shape=(), dtype=string, numpy=b'lay blue in x four now'>]
```

Test on a Video

```
In [61]: sample = load_data(tf.convert_to_tensor('.\\data\\s1\\bras9a.mpg'))

In [62]: print('~'*100, 'REAL TEXT')
[tf.strings.reduce_join([num_to_char(word) for word in sentence]) for sentence in [sample[1]]]
```

Networks for Automated Lip Reading

CHAPTER 5

RESULTS

```
Test on a Video

[266]: sample = load_data(tf.convert_to_tensor('.\\data\\s1\\lbid4p.mpg'))

[267]: yhat = model.predict(tf.expand_dims(sample[0], axis=0))
1/1 [=====] - 0s 186ms/step

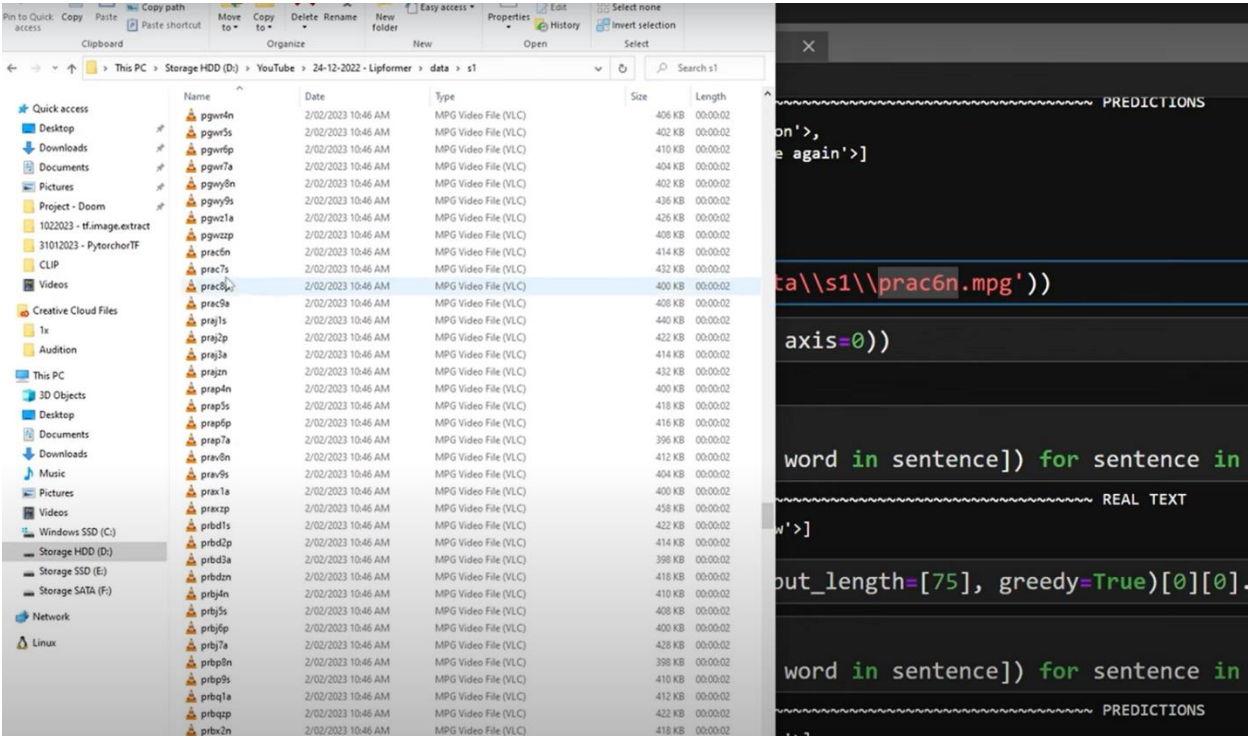
[268]: print('~'*100, 'REAL TEXT')
[tf.strings.reduce_join([num_to_char(word) for word in sentence]) for sentence in
~~~~~ REAL TEXT

[268]: [<tf.Tensor: shape=(), dtype=string, numpy=b'lay blue in d four please'>]

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

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

[265]: [<tf.Tensor: shape=(), dtype=string, numpy=b'set blue in a two please'>]
```



CHAPTER 6

CONCLUSION & FUTURE SCOPE

The conclusion of this research on revolutionizing visual speech recognition through harnessing neural networks highlights several key areas in the application of deep learning. The proposed system integrates Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), specifically leveraging the DeepSpeech2 architecture, to advance visual speech recognition technology. The utilization of CNNs allows for effective feature extraction from visual input data, capturing spatial dependencies and patterns in lip movements. Meanwhile, RNNs, with their ability to model sequential data, complement CNNs by capturing temporal dependencies in speech dynamics. The integration of DeepSpeech2 enhances the system's capabilities by incorporating state-of-the-art techniques in speech recognition, facilitating more accurate transcription of spoken words. In the system design, CNNs process video frames to extract visual features, which are then fed into RNNs for sequence modeling and decoding. The incorporation of DeepSpeech2 further refines the system's performance by providing robust speech recognition capabilities. This comprehensive approach addresses the complexities of visual speech recognition, offering a promising solution for automated lip reading tasks and opening avenues for further advancements in this field.

FUTURE SCOPE

Visual Speech Recognition (VSR), powered by neural networks, is poised for significant advancements in the coming years, driving innovation across various domains. Here are key areas where future developments are anticipated:

1. Enhanced Model Architectures
 - Integration of Transformers: Further exploration of Transformer-based architectures for VSR to capture long-range dependencies more effectively.
 - Attention Mechanisms: Implementing advanced attention mechanisms to focus on relevant facial features during lip reading.
2. Multimodal Integration
 - Audio-Visual Fusion: Advancing techniques to combine visual data from lip movements with audio signals for more robust recognition.
 - Gesture and Facial Expression Recognition: Incorporating gestures and facial expressions to improve context-aware speech recognition.
3. Real-Time Applications
 - Low-Latency Models: Developing models optimized for real-time processing, crucial for applications in live communication and interactive systems.

Networks for Automated Lip Reading

- Edge Computing: Implementing lightweight models suitable for deployment on edge devices, enhancing accessibility and scalability.
- 4. Domain-Specific Adaptation
 - Medical and Healthcare: Customizing VSR models for healthcare applications, such as assisting speech therapy and monitoring patient communication.
 - Industrial and Security: Tailoring models for specific industrial environments and security applications, including surveillance and access control.
- 5. Ethical and Privacy Considerations
 - Data Privacy: Addressing privacy concerns related to facial recognition and speech data in VSR applications.
 - Bias and Fairness: Ensuring models are trained and evaluated to mitigate biases and ensure fairness across diverse demographic groups.

REFERENCES

- [1]. A Lip-Reading Model Using CNN with Batch Normalization published by Harexpe the Gupta, Dhruv Mittal Proceedings of 2018 Eleventh International Conference on Contemporary Computing (IC3), 2-4 August, 2018, Noida, India.
- [2]. Saakshi Bhosale, Rohan Bait, Shivangi Jotshi, Rohan Bangera, Prof. Jinesh Melvin “An Application to Convert Lip Movement into Readable Text” in International Journal of Engineering Research & Technology (IJERT). ISSN: 2278-0181.
- [3]. Souheil Fenghour, Daqing Chen in “Lip Reading Sentences Using Deep Learning with Only Visual Cues” published in November 9, 2020, accepted November 18, 2020, date of publication November 26, 2020, date of current version December 11, 2020.
- [4]. Ahsan Adeel, Mandar Gogate, Amir Hussain, and William M. Whitmer in “Lip-Reading Driven Deep Learning Approach for Speech Enhancement” IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTATIONAL INTELLIGENCE.
- [5]. H. Kulkarni and D. Kirange, "Artificial Intelligence: A Survey on Lip-Reading Techniques," 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2019, pp. 1-5, Doi: 10.1109/ICCCNT45670.2019.8944628

Networks for Automated Lip Reading

[6]. T. Thein and K. M. San, "Lip movements recognition towards an automatic lip-reading system for Myanmar consonants," 2018 12th International Conference on Research Challenges in Information Science(RCIS),2018,pp.1-6, Doi:10.1109/RCIS.2018.8406660.

[7]. W. Nittaya, K. Wetchasit and K. Silanon, "Thai Lip-Reading CAI for Hearing Impairment Student," 2018 Seventh ICT International Student Project Conference (ICTISPC), 2018, pp. 1-4, Doi: 10.1109/ICT-ISPC.2018.8523956