**LIP READING - CORRECTION AGENT**

**Team lead:**

Y. Tejaswini [22B21A4312]

**Team mates:**

K. Manasa [22B21A4304]  
 P. Vardhan [22B21A4333]  
 M. Ganesh [22B21A4335]  
 M. Akshay [22B21A4344]

**Department:** CSE-AI  
**Institution:** Kakinada Institute of Engineering and Technology.  
**Academic Year:** 2025-2026

**Abstract**

Lip reading is the task of recognizing spoken words from visual information of a speaker's lips and facial movements.

This project proposes a Lip Reading + Correction Agent that performs two main functions:  
(a) recognition of words or sub-word units from silent lip videos, and  
(b) correction of these raw predictions into fluent, grammatical full sentences.

The recognition module will use a visual front-end for face and lip detection along with a temporal model such as CNN + Bi-LSTM or Transformer trained on labeled lip-reading datasets.

The expected outcome is a system capable of generating robust word-level recognition from silent video and significantly improving sentence-level accuracy after correction.

Evaluation will be based on metrics such as Word Error Rate (WER), BLEU score, and human judgment of sentence fluency.

**Introduction & Problem Statement**

**Background of the Problem**

Speech perception is naturally multimodal: humans combine auditory and visual cues (lip shape, facial motion) to understand speech, especially in noisy environments. Automated lip reading (visual speech recognition) aims to replicate this ability using machine learning.

Applications include:

* Accessibility (helping hearing-impaired users),
* Silent communication (noisy or privacy-sensitive settings),
* Surveillance, and
* Augmenting noisy speech recognition.

**Why it is Important**

* **Accessibility for the hearing-impaired:** Converts lip movements into text or speech.
* **Improved speech recognition in noise:** Adds a visual channel to improve robustness.
* **Silent communication:** Enables interaction without speaking aloud.
* **Security and surveillance:** Works even when audio is unavailable (e.g., CCTV).

**Clear Definition of the Project / Objectives**

**Project Definition**

The project **“Lip Reading + Correction Agent”** develops an AI system that takes silent video input of a speaker and outputs the correctly formed spoken sentence.

Two stages:

1. **Lip Reading (Recognition Stage):** Extracts visual movements of the lips, predicts raw word/character sequences.
2. **Correction (Language Stage):** Refines predictions into grammatically correct, meaningful sentences using NLP models.

**Objectives**

* Build a two-stage lip reading + correction pipeline.
* Improve recognition accuracy and fluency.
* Combine computer vision + NLP for effective results.
* Evaluate with **WER, CER, BLEU score, and human judgment**.

**Proposed Methodology**

**A. Scratch Model Approach (Designed from Scratch)**

1. **Data Collection**
   * Use datasets: GRID, LRS2, LRS3.
   * Capture **video data** (lip movements), **text transcripts**, and optionally audio.
   * Ensure diversity in speakers, environments, and alignment.
2. **Data Preparation**
   * Extract frames (25–30 fps).
   * Detect & crop mouth ROI.
   * Normalize frames (e.g., 64×64).
   * Align frames with transcripts.
   * Split into train/val/test sets.
3. **Model Design**

* **Visual Encoder (Custom 2D-CNN):**

Instead of 3D-CNN, your project could use a 2D-CNN to process each frame individually and extract features representing the lips’ shape in that frame.

* **Sequence Model (TCN):**

The TCN takes the extracted frame-wise features and models temporal relationships using convolutions over the sequence. It captures how lip movements evolve over time to form words.

* **Decoder (Fully Connected):**

Maps the temporal features to characters or words.

* **Novelty in Your Project:** This approach is fully convolutional (no LSTM or RNN). It allows faster training, handles long sequences efficiently, and is ideal if you want a simpler yet effective model for your correction agent.

1. **Training**
   * Input: Lip frame sequences.
   * Output: Predicted text.
   * **Loss:** CTC or Cross-Entropy.
   * Optimized using backpropagation + GPUs/TPUs.
2. **Evaluation**
   * Metrics: Precision, Recall, Accuracy.
   * Analyze common mistakes.
3. **Hyperparameter Tuning**
   * Tune learning rate, batch size, dropout, model depth, fps.
4. **Deployment**
   * Build an API for real-time webcam/pre-recorded video input.
   * Optimize for low latency & mobile/web use.
   * Feedback loop for continuous learning.

**B. Pre-Existing Models**

Several state-of-the-art models already exist for lip reading:

1. **Lip Net (Assael et al., 2016)**
   * End-to-end sentence-level lip reading.
   * Uses **3D CNN + Bi-GRU + CTC loss**.
   * Demonstrated strong results on GRID corpus.
2. **LCA Net (Xu et al., 2018)**
   * Improves Lip Net using attention mechanisms.
   * Uses cascaded attention and CNN + RNN pipeline.
3. **Visual Speech Transformer (Afouras et al., 2021)**
   * Transformer-based architecture for lip reading.
   * Better long-sequence modeling compared to RNNs.
4. **AV-HuBERT (Shi et al., 2022)**
   * Multi-modal self-supervised learning (audio + video).
   * Achieves SOTA on large-scale lip reading benchmarks (LRS3).
5. **CNN + Bi-LSTM Hybrids**
   * Widely used for lip reading tasks due to strong spatio-temporal modeling.
   * Often serve as baselines for academic and industrial projects.

**Comparison:**

* Pre-existing models provide **high baseline accuracy**, but require large-scale datasets and heavy computation.
* Scratch models allow **flexibility and custom optimization**, useful for smaller datasets or experimental pipelines.

**ML/AI Techniques Used**

* **Computer Vision:** CNN/3D-CNN, MediaPipe, Dlib.
* **Temporal Modeling:** Bi-LSTM, GRU, Transformer.
* **Correction:** Seq2seq or Transformer-based NLP (BERT, T5, BART).
* **Loss:** CTC for recognition, Cross-Entropy for correction.

**Dataset**

* **GRID Corpus** (baseline) – short structured sentences.
* **LRW** – word-level dataset.
* **LRS2/LRS3** – large-scale sentence-level datasets.
* **Synthetic dataset** – noisy → clean mappings for correction.

**Tools and Libraries**

* OpenCV, Media Pipe, Dlib, Pillow, Torch vision.
* Py Torch / TensorFlow.
* Hugging Face Transformers (BART, T5).
* NumPy, Pandas, Py Torch-Lightning.
* Optional TTS: GTTS, pyttsx3.

**Expected Working**

**Input:** Silent video.  
**Process:** Detect lips → extract features → sequence modeling → correction → grammatical sentence.  
**Output:** Corrected text (optionally converted to speech).

**Expected Results**

* Higher accuracy than raw lip reading.
* Fluent, meaningful corrected sentences.
* Applicability in accessibility, silent communication, speech-in-noise.

**Conclusion**

This project integrates **scratch-built lip reading models** and **references to pre-existing models** to design a Lip Reading + Correction Agent.

* Scratch design gives flexibility in architecture and deployment.
* Pre-existing models set performance baselines and inspire improvements.
* The combined methodology ensures robust lip reading and correction, useful for accessibility, surveillance, and silent communication.

**References**

* Chung, J. S., & Zisserman, A. (2016). Lip Reading in the Wild. ACCV.
* Assael, Y. M., et al. (2016). LipNet: End-to-End Sentence-level Lip Reading. arXiv:1611.01599.
* Xu, K., et al. (2018). LCANet: End-to-End Lip Reading with Cascaded Attention-CTC.
* Afouras, T., et al. (2021). Visual Speech Transformers.
* Shi, B., et al. (2022). AV-HuBERT: Self-Supervised Learning for Audio-Visual Speech.
* MediaPipe, Dlib, Hugging Face, PyTorch, TensorFlow.