iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** XII **Month of publication:** December 2025

DOI: https://doi.org/10.22214/ijraset.2025.76076

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XII Dec 2025- Available at www.ijraset.com

MookVaani: An AI-Powered Lip Reader for Real-Time Communication and Speech Generation

Siddhika A. Savant[1], Aliya S. Nadaph[2], Harshada A. Chopade[3], Samruddhi V. Shelake[4], Prof. Amruta M. Kate[5]
[1,2,3,4]*Undergraduate Student, Department of Artificial Intelligence and Machine Learning, Dr. Bapuji Salunkhe Institute of Engineering and Technology*
[5]*Head of Department, Department of Artificial Intelligence and Machine Learning, Dr. Bapuji Salunkhe Institute of Engineering and Technology*

Abstract: ***MookVaani is an artificial intelligence-based lip-reading system, purposed to assist speaking-impaired or those who have difficulty in communicating verbally. Most of the existing speech-recognition tools rely completely on audio input; thus, such tools are useless for non-verbal people or in noisy environments. To bridge this gap, MookVaani picks up the lip movement of a person through a webcam and converts these into text and spoken audio in real time. The system uses MediaPipe for lip landmark detection, while deep-learning models like CNN and LSTM recognize the pattern for visual speech. A Streamlit interface is developed which shows real-time video, predicted text, and generated speech, therefore making the tool easy for any person to use. The project is inspired by the need for inclusive and accessible technology for people with speech disabilities, hearing impairments, or individuals who need silent communication in crowded or sensitive places. MookVaani aspires to provide a method of silent-to-speech communication in an efficient, reliable, and user-friendly way. The system becomes significantly more powerful in sectors like education, healthcare, and accessibility with additional improvements such as multi-lingual support, mobile deployment, and emotion detection.***

Keywords: CNN, LSTM, Lip Reading, Computer Vision, Deep Learning, Speech Generation, Assistive Technology, Human-Computer Interaction

I. INTRODUCTION

Machine learning has rapidly grown to be one of the most influential technologies across modern industries, most especially within the field of artificial intelligence. While AI-driven solutions are still evolving, there is a growing need for systems to support human communication in more inclusive and accessible ways. One of the main challenges that has persisted into the present is the communication barrier between verbal individuals and those who are non-verbal because of speech impairments. Most speech-recognition systems now depend wholly on audio input captured through microphones. While effective for people with clear speech, such systems fail in situations involving silence, unclear vocal output, background noise, or users who cannot produce any sound at all. Due to these disadvantages, a shift towards visual speech recognition is becoming more common, wherein the decoding of communication relies entirely on lip motion and not on sound. Most people cannot naturally understand silent lip motions, as decoding visual speech involves precise pattern recognition that humans are not trained for. This also makes it extremely hard for non-verbal individuals to convey themselves in everyday communication, therefore often resulting in misunderstandings and eventual social isolation. The motivation behind developing a system like MookVaani comes from these real-life struggles. Many non-verbal people have complete cognitive ability and clear thoughts but lack the physical capability to produce sound. This hinders their personal involvement in conversations and results in unease when trying to express themselves. A system that can read lip movements and convert them into text and then speech can greatly reduce this communication gap, and it empowers users with an accessible and effortless means of interaction. The proposed solution intends to provide a real-time, software-based visual speech recognition system capable of processing live webcam input or uploaded video data. By converting the lip movements into comprehensible text and spoken audio, the system offers silent communication with no requirement of any specialized hardware, thus creating a low-cost, user-friendly, and highly inclusive tool suitable for healthcare, education, accessibility solutions, and everyday interaction.

II. PROBLEM STATEMENT

Despite all recent breakthroughs in speech recognition, mainstream systems rely heavily on audio input. This is detrimental to the accessibility needs of users who are non-verbal or have speech impairments. Audio-based systems result in poor performance in noisy conditions, crowded spaces, or situations where the user is unable or is not allowed to speak loudly or clearly enough. In summary, this leads to ineffective support for reliable communication between systems and a large group of users.

Human lip reading is naturally challenging, inconsistent, and requires much training; it is an unrealistic means of communication for most people. Non-verbal people hardly get themselves across because others cannot interpret their silent lip motions. Much of the current visual speech recognition research also falls short, with many systems based on controlled datasets, complicated equipment, or non–real-time processing. These are limiting factors which prevent such technologies from being used in real-world applications.
The core issue at stake is providing an accessible, affordable, and real-time visual speech recognition tool that translates lip movements into coherent text and speech without relying on the use of audio. It is relevant to provide a system capable of functioning through a simple webcam, operating smoothly in real-world environments, and decreasing the communication gap between verbal and non-verbal individuals.

III. OBJECTIVES

The main focus of this work is to develop a visual lip-reading system that would enable silent video of a speaker's lip movements to be accurately converted to textual output. The work will investigate and design a deep learning-based model that learns spatial-temporal patterns of lip motion and recognizes speech visually in a reliable manner.
Specific goals are as follows:

1) Such a deep learning architecture, based on end-to-end training, including CNNs and LSTM, takes the lip region from video inputs and provides word or sentence level text output.
2) Ensure real-world robustness across speakers, accent variations, lighting conditions, head pose, speaking rate, and even background conditions.
3) Create a complete preprocessing pipeline that will include face detection, lip-region extraction, normalization, and data augmentation to improve generalization.
4) Perform the quantitative assessment of system performance using such metrics as WER, sentence accuracy, and speaker-independent generalization. Qualitative evaluation will be performed by visual analysis of prediction patterns and error types.
5) Contemplate some of the possible practical applications of lipreading technology for accessibility, such as for hearing-impaired users, human-computer interaction, and silent or noise-restricted environments.
6) Document methods, limitations, ethical concerns, and suggest future directions by focusing on issues of fairness, privacy, and responsible deployment of visual speech recognition systems.
7) Real-time detection of lip movements with great accuracy.

IV. SCOPE OF PROJECT

This deep learning-based project aims at the design, development, and testing of a visual-only lip-reading system. The scope serves to clearly identify what would be covered in the project and what is excluded or would have to be deferred for research at another time, keeping the workflow realistic and achievable.

A. *Included in Scope:*

1) *Development of a Visual-Only Lip-Reading System*

The project will realize a model that takes silent video input, precisely the face and lip area, and translates it into words or short sentences. The system will solely depend on visual information and not on any help from audio.

2) *Deep Learning Architecture*

The model will use modern techniques in deep learning such as:

- CNNs for the extraction of spatial features from video frames
- Long Short-Term Memory networks, or Transformers for sequential and temporal modeling of lip movements.

The proposed architecture is an end-to-end architecture that jointly learns visual features, time dependencies, and language patterns.

3) *Dataset Usage*

The project will make use of publicly available lip-reading datasets- GRID, LRS1, LRS2. This dataset includes labelled video-text pairs that will enable supervised learning.

4) *Preprocessing and Data Preparation*

It will implement a complete preprocessing pipeline:

- Face and lip region detection
- Cropping video frames
- Frame resizing, contrast adjustments, and illumination normalization
- Data augmentation to simulate realistic variations such as:
  1. Lighting changes
  2. Pose changes
  3. Speaker differences
  4. Background variations

This ensures better model robustness and generalization.

5) Performance Evaluation

The system will be evaluated using standard lipreading metrics such as:

- Word Error Rate (WER)
- Sentence accuracy

Model robustness across:

- Different speakers
- Variable lighting conditions
- Background clutter
- Slight head movements

It will also include qualitative analysis, such as visualizing predictions, misclassifications, etc.

6) Documentation and Reporting

A comprehensive report will be produced covering:

- System design
- Architecture
- Details of the dataset
- Experimental methodology
- Results and limitations
- Ethical considerations: privacy, bias, and fairness
- Future work recommendations

This documentation will be in a research-paper format with proper citations.

B. Outside the Scope (Deferred for Future Work)

Certain tasks are beyond the current scope due to resource, dataset, or time limitations:

1) Lip-Reading in Real Time: Thus, accurate lip-reading directly from a live webcam feed, with real-time processing, requires further optimization, GPU resources, and latency tuning and is hence deferred.
2) Multilingual or Continuous Conversational Lip-Reading: Large, specialized datasets, as well as extended model training beyond current constraints, are required for supporting multiple languages or continuous long-form speech.
3) Audio-Visual Multimodal Systems: This project solely focuses on input through vision, and integrating this system with audio-based speech recognition may be considered for future enhancement.
4) Large-Scale User Studies: Formal usability experiments or testing with hearing-impaired communities are beyond current academic scope and require ethical approvals.
5) Advanced or Specialized Features

Features like these are recognised but left for future consideration:

- Emotion detection
- Speaker identity invariance
- Lip reading under heavy occlusion: masks, hands
- “In-the-wild” unconstrained environment lip-reading

V. LITERATURE REVIEW

A. Overview: Lip Reading

Lip reading is the process whereby spoken language is interpreted based on the observation of the movements of a speaker's lips, facial muscles, and sometimes supplementary facial cues. Humans, notably those who are deaf or hard-of-hearing, lean on lip-reading to understand speech when auditory information is incomplete or unavailable. However, human lip reading has inherent limitations. Many phonemes basic units of sound look similar on the lips, a problem known as the “viseme-to-phoneme” ambiguity. Furthermore, context, familiarity with the speaker, and facial expressions also greatly affect comprehension.

The goal of lip reading, also denoted as Visual Speech Recognition (VSR), is to imitate human lip reading using computer vision and machine learning algorithms. Video inputs of a speaker's face or the lip region are fed into system to generate textual output, phonetic output, or audio output corresponding to the speech content. The early approaches to such systems relied on handcrafted features such as tracking of lip contours, shape descriptors, or geometric representations, combined with classical classifiers like Hidden Markov Models or Gaussian Mixture Models. While performing well in laboratory conditions, these systems did not generalize well to real-world variations in lighting, pose, background, and individual characteristics of the speaker, and were most often vocabulary-limited.

Lip reading has taken a big leap with the introduction of deep learning. While CNNs are responsible for extracting the spatial features from frames, temporal dependencies across multiple frames are handled by RNNs, LSTMs, or Transformer-based architectures. End-to-end models jointly learn both feature extraction and temporal sequence modeling, improving accuracy and robustness. LipNet is one of the earliest well-known end-to-end models demonstrating the feasibility of high-performance automatic lip reading.

B. Recent Advances

Recent lip-reading research has gradually moved to end-to-end deep learning architectures. They aim to minimize explicit feature engineering and enhance generalization across datasets and real-world conditions. We notice several major trends and advances:

1) 3D Convolutional Neural Networks (3D-CNN) + LSTM: These models capture both spatial and temporal features simultaneously. While 3D-CNNs extract spatial features across the frames of a video, LSTMs model the temporal dynamics of lip movement. Many studies have reported high accuracy with word recognition, even reaching as high as 87.5% on controlled datasets. These fusion models handle subtle variations in lip movements better than traditional approaches.
2) Viseme-based Representations: Since most of the phonemes appear similar on the lips, their grouping into "visemes" often helps reduce the classification errors. Such a representation greatly simplifies the mapping from visual cues to phonetic content and improves the recognition rate under noise or uncontrolled conditions.
3) Cross-modal Distillation / Multimodal Learning: Large-scale audio-based speech recognition models are useful in training lip-reading models. The audio models will help the researchers to alleviate the challenge caused by a lack of large annotated video datasets. Techniques such as cross-modal distillation allow the learning of visual speech patterns when video-text paired datasets are scarce.
4) Attention Mechanisms: Some models incorporate attention layers to focus on the most informative regions of the lip and face, dynamically weighting frames carrying key phonetic information. This helps improve performance in longer sequences and complex words.
5) Large-Scale Dataset Utilization: Contemporary research is based on LRW, MIRACL-VC1, and AVLetters datasets that contain data from multiple speakers, accents, and environmental variations. These datasets enable more robust training and evaluation of the models, moving lip reading closer to real-world applicability. Despite these advances, literature surveys and reviews show that automatic lip reading is still error-prone compared to audio-based speech recognition. Challenges in achieving speaker independence, robustness under variable conditions, and real-time deployment remain.

C. Challenges, Limitations, and Open Problems

Even with modern deep learning methods, lip reading faces a number of critical challenges:

1) Visual Problem: Many of the phonemes are mapped to the same or similar-looking lip shapes. For instance, "bat" and "mat" may look virtually identical. It is inherently very difficult to distinguish between such words without additional contextual cues.
2) Speaker Variability: Differences in the shape of the lips, facial hair, skin color, speed, and accents make generalization across speakers difficult. The common scenario is that models, when trained with limited speaker diversity, perform poorly on unseen individuals.

3) Environmental Factors: Performance can significantly degrade with variations in lighting, camera angle, occlusion-such as face masks or hands-and visually noisy backgrounds.
4) Dataset limitations include: The scarcity of large-scale datasets with accurate transcripts of lip movements limits the ability to train models that generalize across languages, speakers, and environments.
5) Performance Gap vs Audio-Based ASR: Even the best performing lip reading systems are far less accurate than their audio-based speech recognition counterparts, since visual information provides only partial cues about speech content.
6) Context Dependence: Human lip readers use not only the lip movements but also facial expressions, gestures, and body language to infer speech. Pure lip-reading systems might miss semantic nuances, especially in continuous speech or emotionally charged conversations.
7) Ethical and Privacy Concerns: Such a powerful lip-reading system might be used for surveillance or unauthorized monitoring. Bias could occur related to gender, skin tone, or accent, raising concerns about fairness and inclusivity.

D. Implications and Research Gaps

The existing literature suggests several avenues for further research:

1) Improving Generalization: Training on varied datasets, domain adaptation, and using self-supervised learning improves the robustness of a model to new speakers and environments.
2) Resolving Viseme Ambiguity: Consider combining visual cues with context in head pose, facial expression, or by surrounding words, and using viseme-based representations to further improve classification accuracy.
3) Multimodal Learning: Cross-modal distillation-pretraining of the visual model using audio-based models-helps alleviate the problem of scarcity of annotated video datasets.
4) Real-world Usability: Beyond benchmark datasets, models should be checked for their performance under uncontrolled lighting, low-resolution cameras, and real conversations. Other practical considerations include latency, privacy, and user acceptance.
5) Multilingual Lip-Reading: Lip-reading research is mostly bound to English. Extending the models to languages like Hindi, Marathi, or any other Indian language with different phonetic and viseme structures is a promising direction.
6) Human-Centered Concerns: Fairness, non-bias, privacy, and consent by users are important considerations, especially if the technology is deployed in public or assistive applications.

VI. PROPOSED METHODOLOGY

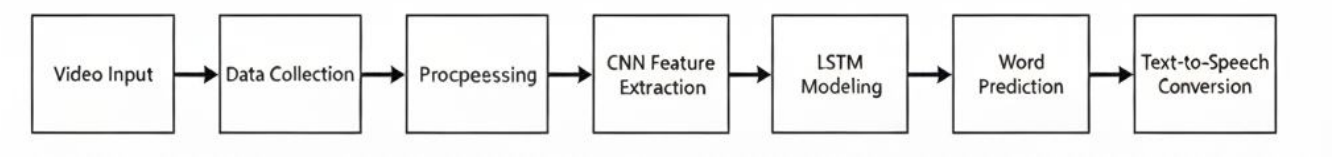


Fig. 1 Flowchart of MookVaani

The proposed system is architectured as an end-to-end visual-only speech recognition pipeline where each stage contributes to transforming silent lip movements into spoken audio output. This methodology includes the following sequential steps:

A. Video Input

It begins with the capture of visual data, which may be through a real-time webcam input or the uploading of a pre-recorded video. Here, raw video captures contiguous lip movements of the speaker. At this stage, the system accepts unprocessed footage, ensuring flexibility and ease of use.

B. Data Collection

In order to train and validate the model, large-scale lip-reading datasets such as GRID or LRS2, LRS3 can be employed. The data consists of aligned video-text pairs, from which the model can learn how visual lip patterns correspond to words that are spoken. Data collection ensures exposure of the system to variability in speakers, lighting conditions, accent, head pose, and speaking style.

C. Preprocessing

The input video starts with preprocessing at various stages to regularize the data:

- Face and Lip Region Detection: Landmark-based detection extracts the mouth region as the primary area of interest.
- Cropping/resizing: the lip region is cropped and then resized to a fixed resolution.
- Frame Normalization: The pixel intensities are normalized to uniform brightness and contrast.
- Frame Sequencing: The video is represented as a sequence of consecutive frames for temporal modeling.

This step eliminates any noise that's irrelevant for vision and prepares orderly data for the model.

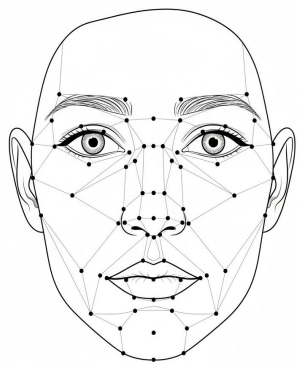


Fig. 2 68-Point Dlib Facial Landmark Detection on a Human Face

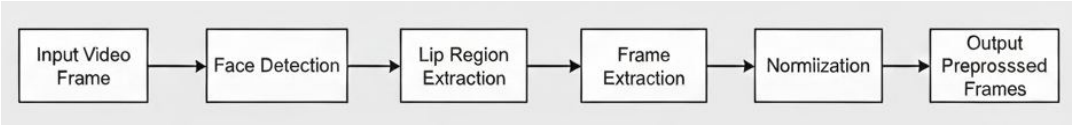


Fig. 3 Detailed preprocessing pipeline diagram

D. CNN Feature Extraction

Each frame obtained from preprocessing passes through a Convolutional Neural Network.

CNN identifies and extracts spatial features such as:

- Lip contours
- Shape variations
- Movement patterns
- Subtle visual cues associated with different phonemes

The CNN transforms every frame into a meaningful feature vector to allow for deeper temporal analysis in the subsequent stages.

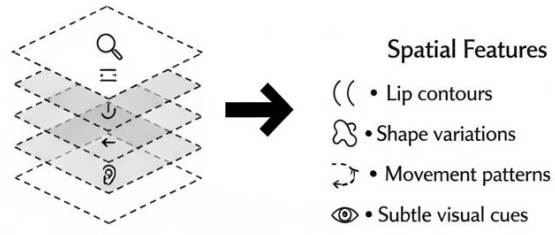


Fig. 4 CNN feature extraction process

E. LSTM Modeling

The sequence of feature vectors is fed into a Long Short-Term Memory network.

LSTM is responsible for capturing temporal dependencies: speech is not static, it unfolds over time.

The LSTM learns:

- How lip shapes transition, frame-to-frame
- Sequential patterns associated with syllables, phonemes, and words
- Timing information encoded in visual articulation

It allows the model to understand whole words from the flow of movement, rather than from isolated frames.

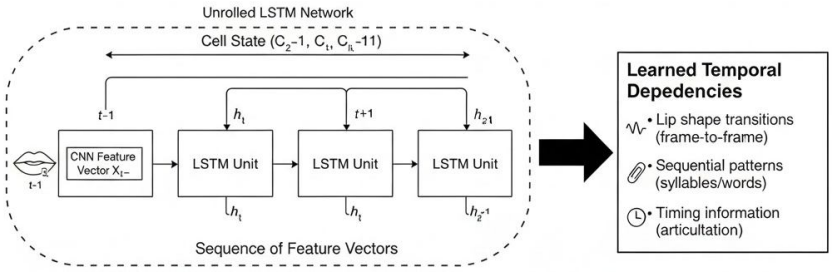


Fig. 4 LSTM sequence modeling

F. Word Prediction

The output of the LSTM layer is passed into a fully connected classification layer.

This layer generates the word or phrase prediction that best matches the observed lip movement sequence. It displays the prediction as the output of visual speech recognition to the user in textual form.

G. Text-To-Speech Conversion

The predicted text is then converted into audible speech using a TTS framework such as pyttsx3. This final step is where the system creates spoken communication, allowing the non-verbal and verbal users to have easy interaction.

VII. EXPECTED OUTCOMES

A. Better Understanding of MookVaani Technology

This project will very effectively demonstrate how a computer system will interpret human lip movements and represent them in textual and audio forms. The data breakdown in understandable steps-data gathering, lip detection, CNN feature extraction, LSTM modeling, and text-to-speech conversion-allows users and students to further grasp the concept of VSR systems. The project not only helps showcase technical workflow but also emphasizes how deep learning can be applied to solve real-world problems.

B. Understanding Communication Challenges

The challenges faced by deaf, hard-of-hearing, and speech-impaired people will come into view through this project. Lip reading is one of the important skills for such people, but human lip reading has operational limits. By studying and implementing MookVaani, we will learn how automated systems can bridge communication gaps, provide independence, and enhance inclusivity in everyday environments.

C. Development of Practical and Useful Technology

The MookVaani system provides a tangible solution to convert silent lip movements into understandable words, sentences, and spoken audio. It thus provides practical applications in noisy classrooms, libraries, public spaces, or workplaces that are restricted in their use of audio. A real-world application of AI in making communication more accessible and efficient is demonstrated in this project.

D. Insights for Future Improvements

The results of the project will shed light on many fronts where the system can be fine-tuned further. This will include, but is not limited to, its predictive accuracy, speed, generalization over different speakers and accents, and robustness across lighting or camera settings. These also will serve as recommendations for future developers and researchers in enhancing the system, paving the way for continuous improvement and adaptation toward increasingly complex real-world settings.

E. Resource for Future Project Developers

MookVaani will serve as a very valuable reference and starting point for students, researchers, and developers who are working on projects involving AI, machine learning, speech recognition, and assistive technology. Such detailed documentation of the methodology and performance evaluation will guide the design of similar systems, selection of appropriate models, and effective integration of various AI components.

F. Guidance for Developers on Models, Tools, and Datasets

Through the implementation and performance analysis of the CNN-LSTM pipeline, study of the pre-processing techniques, and tests on GRID and LRS2 datasets, for example, the result of this project will support developers in selecting the best models, libraries, and tools for lip-reading technologies. This will help streamline future development efforts, reduce experimentation time, and optimize model performance in real-world applications.

G. Educational and Research Contribution

The project exemplifies the application of AI and deep learning to a realistic problem. The MookVaani system can be used by students and researchers to understand better the working of video data processing, feature extraction, training of sequential models, and conversions of prediction into meaningful output. It strengthens knowledge in machine learning, computer vision, signal processing, and human-computer interaction.

H. Impactful Real-World Applications

MookVaani can potentially help people with speech or hearing impairments communicate better and more independently. Besides being an assistive technology, the system can find applications in silent/noise-sensitive environments, security systems for monitoring, accessibility tools for inclusive education, and any other area with limited audio communication. It makes sure that the project has tangible real-life value, beyond being an academic exercise.

VIII. LIMITATIONS OF THE PROPOSED APPROACH

A. Performance Variability in Real-World Conditions

While the system performs with high accuracy under controlled or laboratory conditions, real-world environments introduce a level of unpredictability. Factors such as background noise, varying lighting conditions, unstable internet connections, user errors, or poor-quality video inputs contribute to lowering the responsiveness of the system and predictive accuracy. Real-time performance may become challenged in these scenarios.

B. Dependence on high-quality input data

The quality of video input data is what largely dictates the reliability of this system. Any errors, distortions, lost frames, or low-resolution inputs may greatly hamper the model's correct interpretation of lip movements. Careful data collection, preprocessing, and cleaning are critical in ensuring consistent and accurate outputs.

C. Limited Scalability for Large-Scale Applications

The present architecture has been designed in such a way that it can serve small to medium loads quite efficiently. If the number of users is large or the datasets are huge, it may lead to slow processing, causing latencies. Scaling up the system for large deployments would require extra computational resources, infrastructure, and optimization techniques.

D. Complex Setup and Technical Requirements

Installation and configuration of the system may be tricky for beginners or non-technical people. Its correct setup requires knowledge regarding Python environments, dependencies, deep learning frameworks, and hardware configurations. Without detailed guidance, users might face problems in deployment and operation.

E. Challenges in Integrating Future Features

Major feature enhancements or upgrades may not be supported easily with the existing architecture. For instance, adding new functionality, more models, or advanced capabilities might need complete system redesigns; this could increase the time and cost of development significantly.

F. Security Vulnerabilities

Even though the basic security features were implemented, the system will still be subject to different kinds of risks: unauthorized access, data leakage, malicious inputs, and other forms of cyber-attacks. Assuring data security and system integrity requires continuous monitoring, updating, and strong protective measures.

G. *Hardware and Software Dependencies*

Optimal performance is often directly connected to specific hardware specifications, operating systems, or software versions. If hardware is outdated, incompatible, or low on computational power, system efficiency may decline or the system may cease to work altogether.

H. Ongoing Maintenance Requirements

The system needs continuous maintenance, such as updating models, patching software, performance monitoring, and troubleshooting, to keep the high level of performance going. Performance can degrade or other unexpected technical issues may arise if maintenance is not routinely performed.

I. Limited Handling of Rare or Unseen Scenarios

The system is primarily trained on common datasets and expected patterns of lip movement. It is likely to fail in cases of infrequent, ambiguous, or unconventional input, which may also lead to incorrect word predictions or temporary failure of the system.

J. User Learning Curve

Perhaps the most effective use of the system would involve an investment of time in learning its features and how it works. Poorly understood or badly explained might result in mistakes and/or inefficiency in exploiting the full potentiality of the system.

K. High computational and resource requirements

Real-time video input processing, the execution of CNN and LSTM models, and text-to-speech conversion require a substantial amount of computational resources. For lower-end devices with less memory or computational power, delays, crashes, or suboptimal performance may occur.

L. Limited Cross-Platform Portability

This system may need additional development and adaptation to allow deployment on multiple platforms: web, mobile, or embedded devices. The existing design may not natively support cross-platform use without modification of code.

M. Technical Failures and Downtime

Like any other software system, technical failures are possible. Software bugs, crashes, server downtime, power outages, or other unforeseen hardware/software issues can interrupt system functionality and affect dependability.

N. Cost Implications for Upgrades

System capacity enhancement, advanced feature addition, or hardware upgrades can become expensive. To users on a tight budget, this may also imply limitations to the scaling up of systems and their widespread use.

IX. WORK PLAN

A. Phase 1: Understanding the Problem, Research (Week 1–2)

Week 1:

- Identify the main problem the system solves.
- Do background study on similar tools and existing solutions.
- Document initial project goals and expected outcomes.

Week 2

- Deep research into architecture options (e.g., web-based apps, Replit integration).
- Study frameworks, libraries, and tech stack feasibility.
- Elaborate on high-level system requirements.

B. Phase 2 - Requirement Analysis & Design Planning (Week 3–4)

Week 3

- Define functional and non-functional requirements.
- Prepare the use cases and personas.
- Identify project constraints and assumptions.

Week 4

- Create system architecture diagrams.
- Draw the design flowcharts, data flow diagrams, or sequence diagrams.
- Design UI wireframes and a user's journey.

C. Phase 3: Development Setup & Environment Preparation (Week 5–6)

Week 5

- Setup coding environment on Replit.
- Initialize repository structure.
- Configure dependencies, frameworks, and project folders.

Week 6

- Create base UI screens or initial backend endpoints.
- Implement core planned modules in basic skeleton form.
- Begin the integration of preliminary elements (if necessary).
- Core Implementation & Development

D. Phase 4: Core Development & Implementation (Week 7–9)

Week 7

- Develop main system features - Module 1.
- Test functionality locally on Replit.

Week 8

- Construct further modules: Module 2, Module 3.
- Improve UI responsiveness and refine flow.

Week 9

- Complete the full working version of the system.
- Internal testing of all functionalities.

E. Phase 5: Testing, Evaluation & Optimization (Week 10–11)

Week 10

- Perform functional, usability, and integration testing.
- Bug fixing and optimizations of the codebase.
- Verify the system with sample inputs or user test cases.

Week 11

- Collect feedback from teammates or test users.
- Improve UI/UX and enhance performance.
- Finalize the system behavior based on test results.

F. Phase 6: Final Documentation & Deployment (Week 12)

Week 12

- Compile project documentation: report, diagrams, code explainers.
- Host the final version on Replit.
- Bring the project to real-world.

X. CONCLUSION

This project introduces the steps to be taken in the design and development of a web-based system using modern, available technologies such as Replit. Based on the pace of the workflow, starting from problem analysis to design and implementation and ending with evaluation, the project aims to offer a simple, practical, and scalable solution.

The proposed system will efficiently solve the identified problem but without hampering its ease of use, accessibility, and adaptability to future enhancements. It makes the development process more transparent and easier to manage while structuring the work by clear phases, as done in academic research.

The project will lay a foundation for any application to be stable, reliable, and easy to use. Further extensions can be either for intelligent enhancement or other factors like high performance, possibly with the integration of advanced technology, based on user requirements and practical feedback.

XI. ACKNOWLEDGEMENT

We would like to express our deepest gratitude to all the individuals and institutions who have supported us throughout the development of this project, “MookVaani: A Lip-Reading System Using Deep Learning.” This work would not have been possible without their continuous encouragement, guidance, and cooperation.

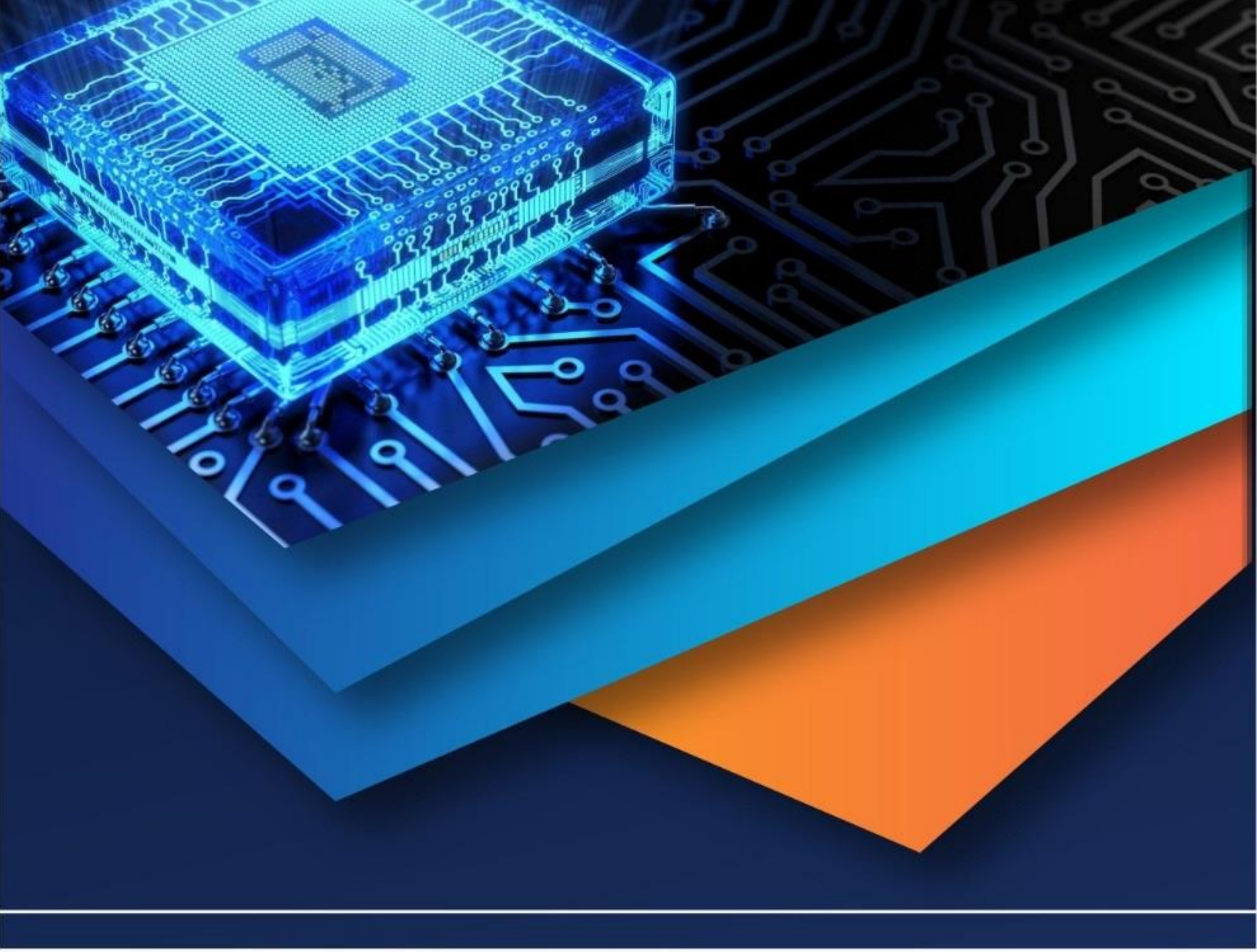
First and foremost, we extend our sincere thanks to our mentors and faculty members, whose expertise and constructive feedback helped us understand the technical, ethical, and practical aspects of implementing an AI-based lip-reading system. Their insightful suggestions shaped our approach at every stage of the research from problem identification and model selection to evaluation and documentation. We are also grateful to the academic environment and resources provided to us, including access to research papers, learning materials, and digital tools. These resources enabled us to gain a deeper understanding of computer vision, neural networks, and real-time inference mechanisms, which form the foundation of this project.

Our heartfelt appreciation goes to all contributing team members. Their hard work, coordination, and willingness to take on responsibilities ensured smooth progress across all phases from requirement gathering and dataset exploration to system design, testing, and report preparation. Their dedication played a significant role in making this project structured, efficient, and meaningful.

We would also like to acknowledge the open-source community and online platforms that provided datasets, pre-trained models, documentation, and technical support. These contributions significantly accelerated our development process and helped us overcome several implementation challenges. Lastly, we thank our families and friends for their patience, motivation, and emotional support throughout the making of this project. Their belief in our abilities kept us consistently inspired and committed to delivering our best. This project is the result of collective effort, guidance, and support from many individuals, and we are sincerely grateful to each one of them.

REFERENCES

[1] M. Wand, J. Koutník, and J. Schmidhuber, “Lip reading with long short-term memory,” IEEE Transactions on Multimedia, vol. 17, no. 11, pp. 1902–1913, Nov. 2015.
[2] Y. Assael, B. Shillingford, S. Whiteson, and N. de Freitas, “LipNet: End-to-End Sentence-Level Lipreading,” arXiv preprint arXiv:1611.01599, 2016.
[3] GRID Corpus. [Online]. Available: https://spandh.dcs.shef.ac.uk/gridcorpus/
[4] LRS2 Dataset. [Online]. Available: http://www.robots.ox.ac.uk/~vgg/data/lip_reading/lrs2.html
[5] OpenCV Library. [Online]. Available: https://opencv.org/
[6] MediaPipe. [Online]. Available: https://developers.google.com/mediapipe
[7] PyTorch. [Online]. Available: https://pytorch.org/
[8] TensorFlow. [Online]. Available: https://www.tensorflow.org/
[9] Pyttsx3 Python Text-to-Speech Library. [Online]. Available: https://pypi.org/project/pyttsx3/
[10] Replit. [Online]. Available: https://replit.com/
[11] Next.js Documentation. [Online]. Available: https://nextjs.org/docs
[12] J. Brownlee, Deep Learning for Time Series Forecasting: Predict the Future with MLPs, CNNs and LSTMs in Python, Machine Learning Mastery, 2018.
[13] N. Deshmukh, A. Ahire, S. H. Bhandari, A. Mali, and K. Warkari, "Vision based Lip Reading System using Deep Learning," in *2021 International Conference on Computing, Communication and Green Engineering (CCGE)*, Pune, India [Online].Available: https://www.researchgate.net/publication/362015222_Vision_based_Lip_Reading_System_using_Deep_Learning



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY