OPEN ACCESS

ADVANCES IN COMPUTER SCIENCE AND ENGINEERING

https://sci.pphmjopenaccess.com/index.php/acse
Volume 20, Number 1, 2026, Pages 1-20
https://doi.org/10.17654/0973699926001

P-ISSN
0973-6999

VoiceBridge: AN AI-POWERED FRAMEWORK FOR LOW-COST MULTILINGUAL VIDEO DUBBING INTO INDIAN REGIONAL LANGUAGES

M. V. Madhusudhan, Annmary Jojo, D. Aqsa Mehreen and H. S. Agampreeth

Department of CSE, Presidency University, Bengaluru, India
e-mail: mv.madhu@gmail.com; annmary.20221CSG0140@presidencyuniversity.in; aqsa.20221CSG0118@presidencyuniversity.in; agampreeth.20221CSG0117@presidencyuniversity.in

Abstract

VoiceBridge is an AI-powered, low-cost multilingual dubbing framework specifically built to make English video content available in various Indian regional languages, specifically South Indian languages: Kannada, Tamil, Telugu, and Malayalam. VoiceBridge combines bleeding-edge open-source technologies such as Whisper for Automatic Speech Recognition (ASR), IndicTrans2 for Text Translation (TT), and Coqui or Indic-TTS for Text-to-Speech (TTS), to create an end-to-end pipeline of transcription, translation, speech synthesis, and video dubbing that is affordable, culturally relevant, and easily scalable. The framework features a simplified interface that allows users to upload videos, translate speech, and produce dubbed outputs without having to have any background knowledge of the processes or technologies being used. Evaluation of performance gave promising results, arriving at a Word Error Rate (WER) of 11.9% and Character Error Rate (CER) of 11.09%, showing significant levels of recognition and translation accuracy despite minor differences in pronunciation and a 90millisecond audio-video delay. VoiceBridge utilizes open-source models and adapts those for low-resource languages to serve as abridge towards mitigating the digital languages gap, and as a means to provide access to educational and informational content in video format to various linguistic communities.

Keywords and phrases: multilingual, artificial intelligence, word error rate, character error rate, accuracy, whisper model.

Received: November 3, 2025; **Accepted:** November 24, 2025;

Published: December 8, 2025

1. Introduction

In the last decade, the way people consume knowledge and information has undergone a dramatic transformation. With the rapid growth of the internet and digital technologies, video-based content has become one of the most powerful and engaging mediums for communication, education, and entertainment. Online platforms such as YouTube, Coursera, edX, and Khan Academy have brought learning out of the confines of the traditional classroom and made it accessible to millions of learners worldwide [1]. The increasing penetration of affordable smartphones and high-speed internet in many of the developed and developing countries like India has further accelerated this shift. For many learners, especially in remote or rural areas, videos offer the first and sometimes the only exposure to structured learning resources. Compared to offline lectures and printed material, videos offer several advantages: they are reusable, they allow learners to pause and re-watch difficult concepts, and they combine both audio and visual cues to enhance retention. In short, online video content has become indispensable to modern education and information dissemination. A UNESCO survey of 61 countries found that 90% of high-income education systems rapidly adopted online learning platforms during school closures-showcasing the nimble adaptation of digital infrastructure under crisis conditions [2].

Despite this global revolution in digital learning, language continues to remain a significant barrier. The vast majority of educational and informational videos available online are in English or a handful of other widely spoken global languages [3]. While this benefits a large international audience, it excludes a massive segment of the population in countries like India, where hundreds of millions of people are more comfortable in their native tongues. English dominates the World Wide Web (WWW) as around 62.5% of the material available on the Internet is written in this language [4]. This problem is part of a larger digital divide, as over 80% of online content is available in just 10 languages creating a significant information access gap for speakers of the world's remaining 7,000 languages [5]. According to the 2011 census of India, only 10.67% of the total population spoke English, of which only 0.02% had it as their first language [6]. In 2023, approximately 43.4% of the Indian population was using the Internet, and this figure is expected to rise to around 62.8% in the coming years [7]. This statistic reveals a gap in the amount of available content on the internet and the number of Indians who can access it [8]. India is not just a linguistically diverse nation, it is one of the most linguistically rich countries in the world, with 22 officially recognized languages and hundreds of regional dialects [9]. Although the digital landscape is rapidly expanding, English continues to maintain its control of the digital space while Indian languages, especially South Indian languages like Tamil, Telugu, Kannada, and Malayalam, are not receiving corresponding attention. People relate to an area with which they have deeper affective and cognitive connections when the information is available in the local tongue-the mother-tongue closest to their heart [10]. Therefore, an overwhelming amount of available online content that is valuable for students, professionals, and society as a whole remains locked away for those who cannot engage with English.

The lack of global digital visibility and representation for South Indian languages now is a significant issue. With tens of millions of speakers, they have rich cultural, literary, and historical heritages. However, in computing linguistics, South Indian languages are categorized as low-resource languages. Low-resource languages receive less research attention and commercial investment than high-resource languages like English, Spanish, or Mandarin. This digital divide excludes speakers of South Indian languages and prevents participation in modern knowledge systems, global discourses, or the benefits of the digital world. Therefore, bridging this gap is not a technological challenge, but a social obligation.

Numerous solutions have emerged over the years to solve the language access issue in the context of video content. Commercial services such as Papercup, Veritone, Synthesia, and HeyGen leverage artificial intelligence to provide dubbing and voice-over for content in a variety of languages, while YouTube has also piloted automatic subtitling as well as limited support for community translation into other languages. These programs certainly represent a significant technological advancement - but they exhibit not insignificant deficiencies from the lens of Indian languages and the question of affordability. First, many of the these systems charge subscription fees that are too expensive for smaller institutions, NGOs, educators, and individuals that cannot afford enterprise pricing. Second, their language coverage is primarily biased toward high-resource Western (and to some extent East Asian) languages. South Indian languages, in particular, either lack support entirely, or if they do receive support, their quality is often low, resulting in translations that can have very poor accuracy and/or awkward phrasing. Third, a lot of these systems struggle to capture cultural context. Even where the literal translation works grammatically, the message when delivered as a spoken message might sound unnatural, or worse, confusing, to audiences in a regional language. All of these issues are particularly problematic for education: being clear and contextually aligned is critical for learning.

Another highlighted limitation of existing dubbing products is that they are proprietary and closed source. Most existing dubbing platforms are black boxes without insight for users or researchers into their processes, that does not allow, retraining models, or customizing for specific purposes. This might limit possibilities for innovation and use in local contexts without transparency and customisation options. The platform proposed supports a number of video types, meeting different user needs from short clips to longer documentaries [11]. In addition, many existing platforms are cloud-based and depend on having stable internet with high-performance bandwidth. In rural and semi-urban areas in India, providing such services remains challenging, and so even with modern technological approaches, most current dubbing solutions are failing in fishable reaching or benefitting the talent and communities that need it most.

Our proposed system aims to overcome these limitations by focusing specifically on the needs of Indian regional languages, with a particular emphasis on South Indian languages such as Kannada, Tamil, Telugu, and Malayalam. Unlike commercial platforms that are designed for global corporate clients, our system is designed to be affordable, accessible, and inclusive. It uses open-source models like Whisper for ASR, MarianMT for machine translation, and Coqui TTS for text-to-speech synthesis [12]. By combining these freely available resources, we create a pipeline that can take an English-language video as input and produce a dubbed version in the target Indian language as output. Importantly, this approach is cost-effective, requiring only modest computational resources and a budget well within the reach of student projects or NGOs. This makes our system unique to address real-world problems in education and awareness campaigns, especially in cases with low funds.

Another unique characteristic of our proposed system is the cultural and semantic accuracy focus in the applications. We are not strictly translating on a word-by-word-basis-we are attempting to maintain the meaning and context within the source material and target language [13]. This increases the understandability and relatability of the dubbed videos, making them more effective as teaching and communication tools for their intended audiences. Furthermore, our design considers ease of use, so specifying a system as a simple, web-based hosted platform has added functionally even to non-technical users. Users only need to upload a video file to the system, select the desired target language, and download the dubbed version. There is no need to have special skills or configurations. This web interface, as well as providing offline and low bandwidth options, enhance system usability in both urban and rural environments.

Along with addressing practical problems, our research work also is a contribution to academia and research in the field of natural languages processing and speech technology [14]. By applying and adapting existing open-source AI models to low-resource languages, the system demonstrates how cutting-edge AI can be localized and democratized [15]. This not only helps communities that are otherwise excluded from the digital revolution but also enriches the research ecosystem by providing new insights into handling underrepresented languages. These audios are divided into segments to obtain relevant features. Unlike global commercial solutions, our project emphasizes openness, customization, and social impact.

In summary, the rapid growth of online video learning has created new opportunities but also highlighted old barriers. Language remains one of the most significant challenges to true digital inclusivity. While existing dubbing tools have made progress, their focus on high-resource languages, high costs, and lack of cultural sensitivity limit their usefulness in contexts like India. Our proposed method, by contrast, is unique in its focus on affordability, inclusivity, and cultural relevance, with special attention to South Indian languages. By building a system that is open-source, scalable, and user-friendly, we aim to contribute both to the academic research community and to the broader social goal of making knowledge more universally accessible. In doing so, this research work not only addresses a pressing technical challenge but also responds to the larger vision of creating a more equitable and linguistically inclusive digital future.

Flowchart/Block diagram

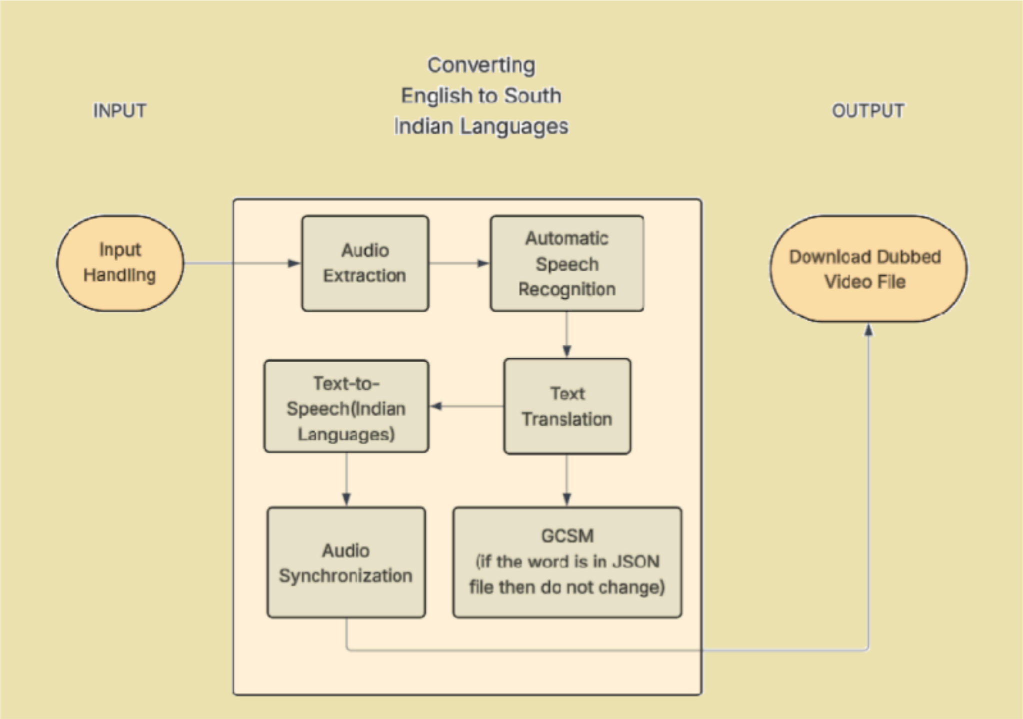

INPUT
Converting English to South Indian Languages
OUTPUT
Input Handling
Audio Extraction
Automatic Speech Recognition
Text Translation
Text-to-Speech(Indian Languages)
Audio Synchronization
GCSM (if the word is in JSON file then do not change)
Download Dubbed Video File

Figure 1. Block diagram of processing pipeline for video dubbing in Indian languages.

2. Literature Review

In recent years, researchers have explored multiple approaches for automatic video dubbing, combining speech recognition, translation, synthesis, and audiovisual alignment. Zhang et al. proposed a generative AI-driven multilingual dubbing and synthesis system that integrates ASR, TT, and TTS into a unified workflow [16]. Their methodology centred on modular pipeline implementation with runtime evaluation, but the system faced drawbacks such as high real-time latency and limited prosody preservation. Evaluation was done through throughput and subjective naturalness, and they identified future work in handling low-resource languages and improving real-time adaptation. Similarly, Li et al. designed a multimodal TTS with multi-scale style control for dubbing [17]. The methodology introduced hierarchical style encoders for expressive prosody. While MOS evaluations confirmed naturalness gains, the main drawback was inconsistent cross-speaker voice cloning. Future improvements were suggested for prosody alignment and lip synchronization.

For direct speech-to-speech solutions, Chen et al. developed end-to-end spectrogram-to-spectrogram audio style transformation models [18]. The methodology by passed TT by learning spectral mappings. Evaluation used spectral similarity and intelligibility scores, but drawbacks included poor prosody transfer and limited multilingual datasets. Future prospects include building large aligned corpora and better speaker identity preservation.

On the visual side, Rahman et al. introduced "Seeing the Sound," a real-time lip-sync system for multilingual dubbing [19]. Their methodology combined TTS with neural lip-synthesis models, evaluated with lip-sync error metrics and user perception tests. However, drawbacks included inaccurate phoneme-to-viseme mapping in cross-language settings. They suggested future research on improving mapping models and robustness under diverse head poses. Similarly, Gupta et al. stabilized talking-face generation by introducing new training losses [20]. Their methodology reduced lip-identity leakage and was evaluated using visual fidelity and sync scores. Despite improvements, challenges remained in generalizing to natural, in-the-wild scenarios. Wang et al. further contributed with Style Sync-like systems, mapping target audio to facial motion [21]. Evaluation involved sync accuracy and perceptual preference tests, though drawbacks included poor co-articulation across languages. Future directions point to context-aware viseme synthesis. Lee et al. leveraged vision transformers for audiovisual speech synthesis [22]. Their methodology improved lip-sync and expression alignment, validated with visual fidelity metrics, but incurred high computational costs, suggesting optimization as a future step. Zhou et al. advanced diffusion-based video editing methods for dubbing, producing coherent lip-sync edits, but the drawback was heavy computation [23]. Evaluation included temporal coherence and lip-sync accuracy, with future work aimed at lightweight diffusion architectures.

Evaluation methods themselves have been refined. Martinez et al. proposed PEAVS, a perceptual metric for audio-visual synchrony [24]. Their methodology combined metric design with large-scale perceptual validation. The drawback was limited testing on multilingual dubbing datasets. Future improvements include adaptation to diverse phonetic structures. In their review Alonso et al. assessed methodologies within deep learning models, comparing accuracy and error rates while pointing to robustness limitations within edge cases such as noise and cross-language [25]. Future directions include establishing and employing comprehensive multimodal fusion integrations filters into dubbing workflows. In similar vein, Santos et al. evaluated continuous Spanish lipreading using hybrid end-to-end CTC/attention models demonstrating strong WER/CER, while others limited evaluations to Spanish [26]. They identified an area for future use case to expand evaluation into multiple languages.

In addition to the focused work around technology, Muller et al. reviewed multilingual dubbing systems in the paper under consideration and identified a taxonomy along with a description of gaps reviewed multilingual dubbing systems, presenting a taxonomy and identifying gaps [27]. Their methodology was systematic literature analysis, with the drawback of lacking empirical validation. They stressed the need for standardized multimodal datasets as a future direction. Ferndez et al. analysed dubbing and subtitling strategies using comparative user studies [28]. Evaluation focused on comprehension and cultural adaption, though small sample sizes limited generalizability. They suggested integrating automated dubbing into cross-cultural studies. From an educational point of view, Kumar et al. found that dubbing activities enhanced pronunciation among non-English speakers, while Wang et al. [29]. Observed that it helped lower learner's anxiety during speaking tasks [30]. Both used classroom interventions, including surveys and pre- and post- tests. Limitations included small cohorts and subjective scoring, with future recommendations for solid digital dubbing platforms for education. Lastly, Singh et al. raised ethical concerns on deepfake detection methods [31]. Their method included a review of detection algorithms - tested in benchmarks. Limitations included susceptibility to adversarial attacks and future work relating to watermarking and source tools supporting safer adoption of dubbing.

Overall, methodologies across these works span modular pipelines, direct end-to-end audio style models, audio-driven talking-head generation, and novel perceptual metrics. Common drawbacks include limited datasets, phoneme-viseme mismatches, inadequate prosody transfer, high computational costs, and ethical risks. Evaluation typically combines MOS, WER, sync error metrics, spectral similarity, and human perception studies. The consensus across studies is the need for richer multilingual datasets, lightweight real-time models, and ethical safeguards, with particular emphasis on expanding dubbing technologies to underrepresented languages such as Indian regional dialects.

3. Proposed Methodology

Our proposed video dubbing system follows a structured and modular pipeline designed to automatically translate and dub English videos into Indian regional languages with high, linguistic coherence, and cultural sensitivity. The methodology is divided into several interconnected components: Input Handling, Automatic Speech Recognition (ASR), Text Translation (TT), Glossary and Code-Switching, Text-to-Speech (TTS), Audio-Video Synchronization, and Output Handling.

3.1. Input handling

The pipeline initiates with an input handling module, in which users upload video files in .mp4 file format, through a user-friendly web interface. The web interface is developed in a combination of HTML/CSS and JavaScript to improve user experience due to it being more dynamic. The backend uses Python frameworks like Flask to facilitate uploading files to the server. Flask is an option to enable rapid prototyping and synchronous HTTP request handling, while FastAPI can either use synchronous or asynchronous file handling to improve processing performance when uploading multiple files concurrently. Uploaded video files are sent to the server for temporary storage, and saved with secure file handling mechanisms. For file handling, Flask uses the Werkzeug library, and FastAPI uses audio files for asynchronous writing of large video files to disk. Once verification of a successful upload of the video file has been performed, the video is prepared for the next step in the processing pipeline.

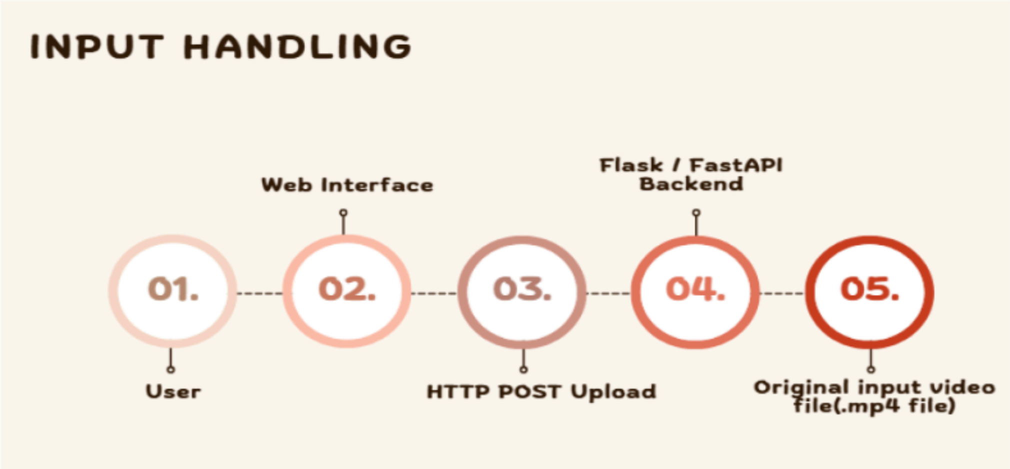


Figure 2. Input handling workflow.

3.2. Audio extraction

Audio extraction is performed by isolating the audio stream from the video file (input). By using a multimedia processing tool such as ffmpeg, the system removes the video and audio components and converts the audio into a standard WAV format at the 16kHz sampling rate with mono channel, which is more compatible with ASR models. This means that the audio was extracted from the video and is ultimately cleaned and consistent to various video types. Before the audio is fed into the speech recognition unit, some initial preprocessing - including noise reduction and normalization - can be applied to increase the audio quality overall.

$$V_i = V(f) + A(t),$$

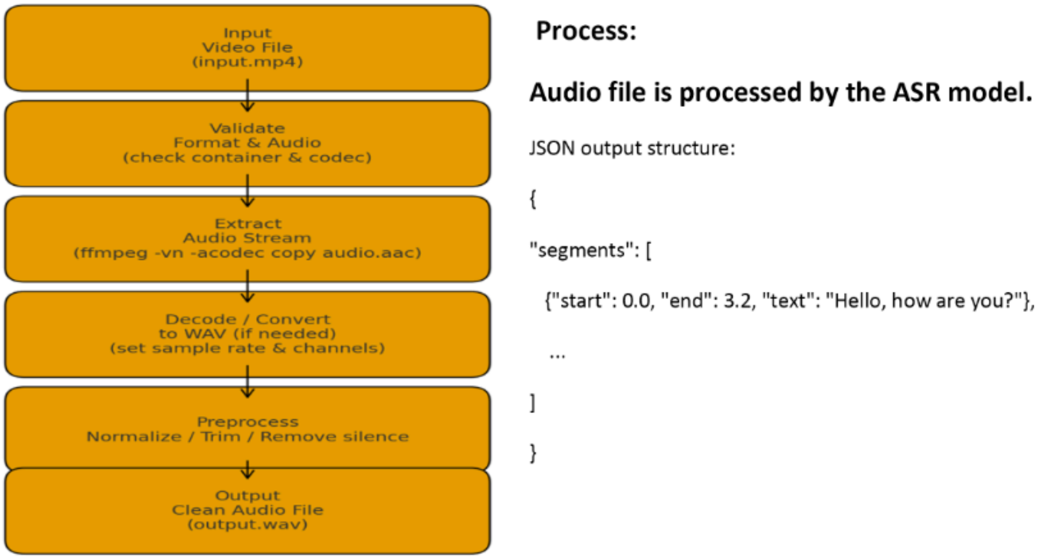
$$A_o = Extract(V_i) = A(t). \tag{1}$$

In equation (1), the input video V_i has visual frames $V(f)$ which may be different from two audio signals $A(t)$. In this case, $V(f)$ represents the video component that changes defined by a frame rate f, while $A(t)$ is audio that changes defined by a frame rate f, while

$A(t)$ is audio that exhibits similar differences over a time scale. Extract (V_i) extracts the audio part of the video, and A_o is the audio extracted from video used for further processing such as speech recognition.

3.3. Automatic speech recognition (ASR)

The ASR component is intended to extract the spoken dialogue in English from the uploaded video and translate it into a text transcript with timestamps. This allows for precise alignment of the spoken content with the video timeline for further processing and analysis. The audio stream is first extracted from the video using the ffmpeg-python library, which maintains the input video as a .wav audio file. There are two options for ASR model choice to generate the transcript, OpenAI Whisper and Vosk. OpenAI Whisper generates high levels of transcribing ability across various accents of English, and it generates a structured JSON containing the transcript and specific level accuracy in word timestamps. Vosk is intended for an offline version of ASR, useful if working in environments without connectivity. The ASR model processes the audio extracted from the video, and creates a structured transcript output in JSON format as an array of segments, where each segment contains a start timestamp, end timestamp, and associated text. This output can be used as a source of diversion for subsequent processing treatment for alignment.



Process:

Audio file is processed by the ASR model.

JSON output structure:

```
{
"segments": [
  {"start": 0.0, "end": 3.2, "text": "Hello, how are you?"},
  ...
]
}
```

Figure 3. ASR flowchart.

3.4. Text translation (TT)

Following the transcription, the system translates the English transcript into the desired Indian regional language. We primarily use IndicTrans2, developed by AI4Bharat, which is a state-of-the-art text translation model specifically optimized for Indian languages. As a fallback, MarianMT from Hugging Face Transformers can be used. The translation process involves passing individual sentences from the English transcript into the TT model using the transformers library. Special attention is paid to sentence segmentation to avoid splitting sentences improperly, which could compromise translation accuracy. The translation model outputs the target language text while maintaining alignment with the original timestamps. Mathematically, this process is represented by the equation is the English transcript, and the translated regional language text. The translation equation (ii) is given below where T_e refer to English transcript and T_r refer to Translate text in regional language. The result is a translated transcript file containing sentences in the target language with corresponding timestamp. The process begins by taking English sentences and passing them to a translation model built using Hugging Face's Transformers library. The model translates each sentence into the target language while preserving the meaning and context. Along with the translation, timestamps are generated to align each translated sentence with the corresponding segment of the original content. This ensures that the translated text is synchronized with the timing of the source material, making it suitable for applications such as subtitles, dubbing, or real-time multilingual communication.

$$T_r = MT(T_e). \tag{2}$$

In equation (2), the English text (T_e) is first translated using the translation model (T), and then this output is refined or mapped by M to produce the final regional translation (T_r).

3.5. Glossary and code-switching module (GCSM)

A key novelty of our methodology lies in the Glossary and Code-Switching module, designed to prevent mistranslation of technical, cultural, or proper noun terms. A predefined JSON dictionary stores terms that must remain unchanged during translation, such as acronyms ("GST", "AI"), proper nouns, or platform names ("YouTube"). During translation, each word or phrase in the transcript is scanned using Python's regex module to detect matches against the glossary dictionary. If a word exists in the glossary, it is preserved as-is, bypassing the translation process. This ensures that important terms remain intact in the translated output, where DD is the set of glossary terms and ww is a word from the transcript. This module significantly improves the quality of the translated script by maintaining cultural and technical accuracy, avoiding the common problem of mistranslation in standard TT workflows. The final regional translation (T_r') is given by equation (3):

$$T_r' = \begin{cases} T_e & \text{if } \omega \in D \\ MT(\omega) & \text{otherwise.} \end{cases} \tag{3}$$

In equation (3), if the word ω exists in the Domain Mapping Table (DMT), then final regional translation (T'_r) keeps the English translation (T_e). Otherwise, it follows the normal regional translation process.

3.6. Text-to-speech (TTS)

The TTS module converts the translated text into a natural-sounding audio file in the target regional language. We utilize either Coqui TTS or Indic-TTS from the IIT Madras project, both of which offer support for Indian languages and multiple voice options. For each sentence in the translated script, the TTS engine generates a corresponding audio segment. The process can be represented by the formula. Voice parameters such as pitch, speed, and intonation are adjustable to enhance the naturalness of the speech. The generated audio segments are concatenated in the correct order, respecting the original transcript's timestamp alignment. The final output is a seamless regional language audio file (.wav or .mp3), ready for integration back into the video. For each Sentence S_i audio output is generated using the equation (4):

$$A_i = TTS(S_i). \tag{4}$$

In equation (4), The audio output A_i is generated by applying Text-to-Speech (TTS) to the sentence or text segment S_i. In simple terms, S_i is converted into spoken audio using a TTS system.

3.7. Audio-video synchronization

Input: Original video file (input.mp4)

Output: Dubbed video file (dubbed.mp4)

```
load_files (video, audio)
video = load_video("input.mp4")//Load original video file
audio = load_audio("generated_audio.wav")//Load generated audio file
```

In the audio-video synchronization stage, the newly generated audio is merged back into the original video to produce a dubbed version. The ffmpeg-python library is used to replace the original English audio stream with the generated regional language audio, preserving video quality. The merging operation follows the function given below:

```
merge (audio, video):
{
    dubbed_video = merge_audio_video (video, audio)
    save_video (dubbed_video, "dubbed.mp4")
}
```

For enhanced realism, the system optionally integrates the Wav2Lip model, which adjusts the speaker's lip movements to synchronize with the dubbed audio. The Wav2Lip model takes the original video frames and the generated audio as input, modifying the lip region of each frame using a deep learning algorithm. The lip-synced frames are then reassembled into the video stream, and the new audio is integrated. This step ensures that the final video appears natural and immersive for the viewer.

3.8. Output handling

After the dubbing process has been completed, the user will have a way to download the final .mp4 video file. This is accomplished through a download endpoint that is implemented using Flask. Once each file has been processed, it's saved on the server under a unique session ID to prevent multiple users from overwriting files at the same time and stored for file retrieval.

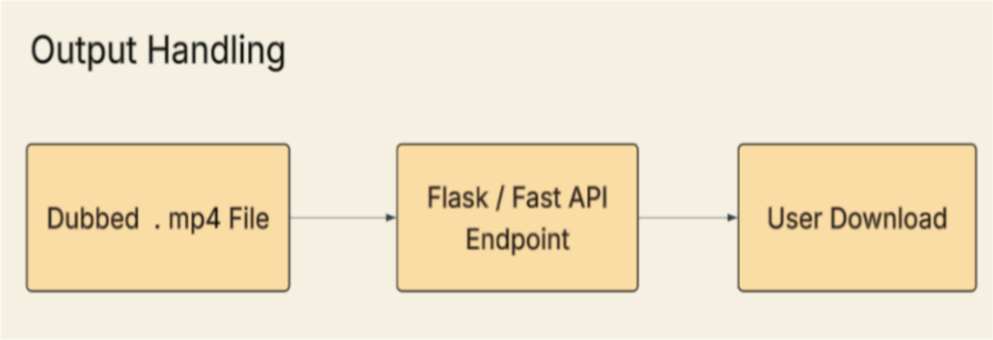


Figure 4. Output handling workflow.

4. Result and Discussion

The VoiceBridge framework, as presented, provides an affordable, time, and AI vection to support multilingual video dubbing in Indian regional languages, focusing on access, affordability and language accuracy. Our proposed framework integrates multiple open-source tools and models into a modular pipeline that translate and dub English video content to regional languages, such as Kannada, Telugu, Tamil, and Malayalam. The clips of the video content in English must be translated and dubbed in regional languages by leveraging the extensive set of open-source tools and models in Python. The necessary tools and models include ffmpeg-python for audio/video extraction and merging, Flask for backend processing and uploading via the web, Hugging Face Transformers for Text Translation (TT) via IndicTrans2 and MarianMT, (both of which fine-tuned on Indian definitions), and the Automatic Speech Recognition (ASR) component uses OpenAI Whisper and Vosk (both of which provide reliable and accurate transcripts for a range of English accents). The Text-to-Speech (TTS) is based on using Coqui TTS and Indic-TTS to produce smooth natural-sounding voiceovers in the regional voices. Optionally, dubbed audio can be synced with lip movements using a Wav2Lip model for realism and quality.

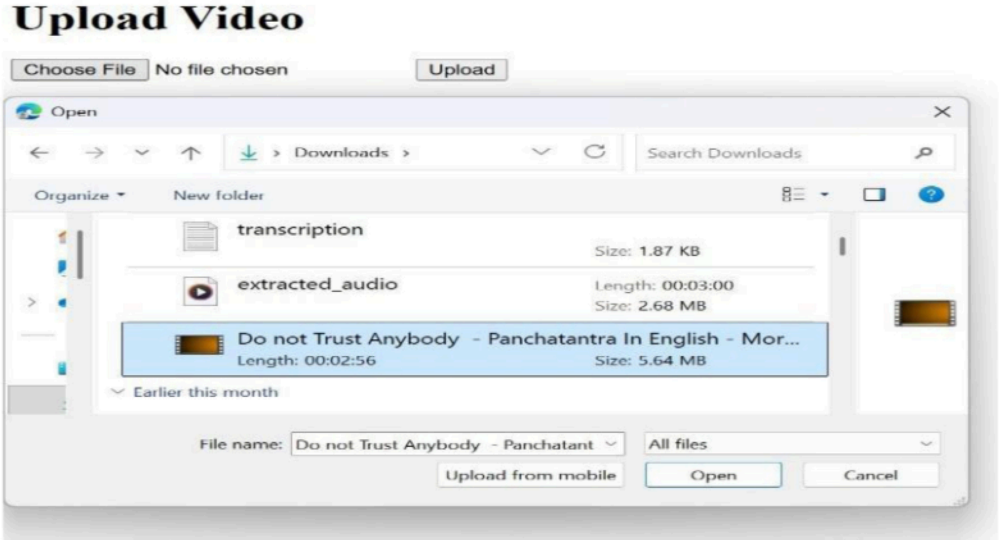
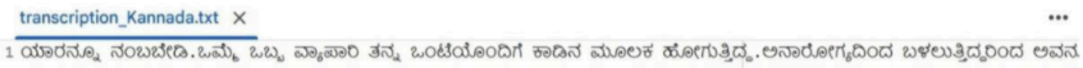


Figure 5. Input video upload.

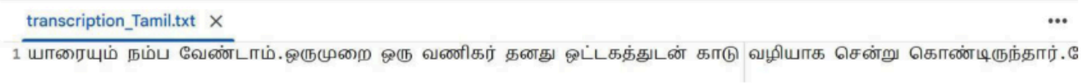
This video dubbing program makes it simple to convert English videos into South Indian regional languages. Users can upload a video, select target languages, and turn on all features for automatic dubbing. The system recognizes the sound, translates the video, and creates a voice. A fast, functional, and accurate automated dubbing system will give users an excellent experience in multiple languages. The image shows the user uploading a video file to the dubbing software. The software interface allows a user to pick a video file to translate into a dubbing experience in South Indian languages. The first step is to extract track from the video using specialized audio processing software tools. This will separate the audio components of speech, music, and background sounds into an appropriate medium like a .wav/.mp3 file. By extracting the audio track, it can be precisely deciphered into scripted text without the subsequent noise created by visual mediums. After audio extraction, the audio can be set up for ASR automated next steps, or, it can be transcribed to be translated for future dubbing efforts, where video dubbing can be done more efficiently and accurately. The extracted audio can then be processed for ASR or Spoken Document Processing (SDP). ASR assesses the audio elements, recognizes spoken words via acoustic analysis, and builds an accurately sequenced text from the initially undetermined spoken audio.

Output in the Kannada Language.



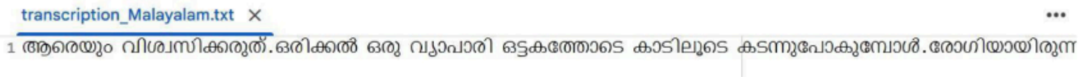
```
ಯಾರನ್ನೂ ನಂಬಬೇಡಿ.ಒಮ್ಮೆ ಒಬ್ಬ ವ್ಯಾಪಾರಿ ತನ್ನ ಒಂಟೆಯೊಂದಿಗೆ ಕಾಡಿನ ಮೂಲಕ ಹೋಗುತ್ತಿದ್ದ.ಅನಾರೋಗ್ಯದಿಂದ ಬಳಲುತ್ತಿದ್ದರಿಂದ ಅವನ
```

Output in the Tamil Language.



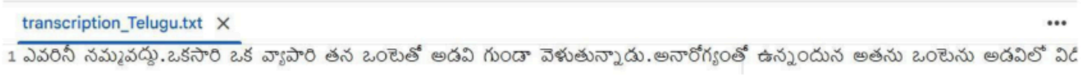
யாரையும் நம்ப வேண்டாம்.ஒருமுறை ஒரு வணிகர் தனது ஒட்டகத்துடன் காடு வழியாக சென்று கொண்டிருந்தார்.

Output in the Malayalam Language.



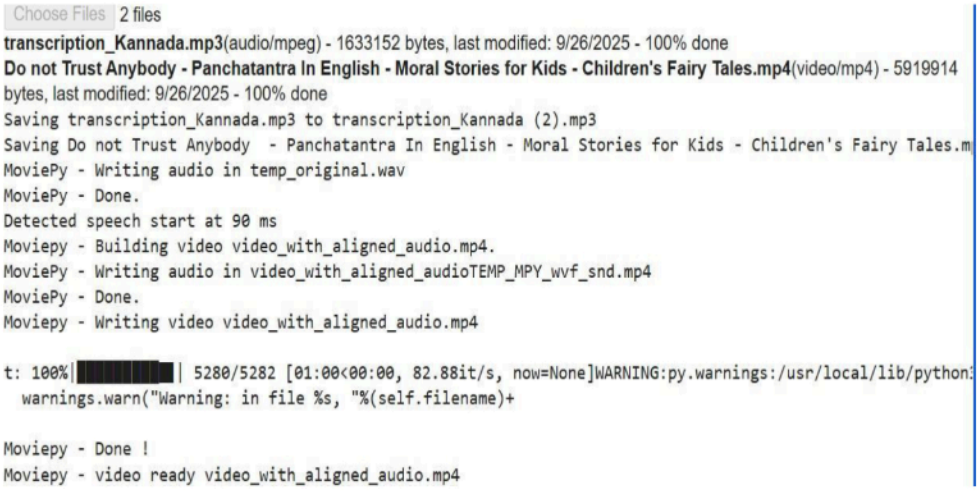
ആരെയും വിശ്വസിക്കരുത്.ഒരിക്കൽ ഒരു വ്യാപാരി ഒട്ടകത്തോടെ കാടിലൂടെ കടന്നുപോകുമ്പോൾ.രോഗിയായിരുന്ന

Output in the Telugu Language.



ఎవరినీ నమ్మవద్దు.ఒకసారి ఒక వ్యాపారి తన ఒంటెతో అడవి గుండా వెళుతున్నాడు.అనారోగ్యంతో ఉన్నందున అతను ఒంటెను అడవిలో విడ

The text that has been transcribed can also serve for functions such as translation, dubbing, or subtitle generation. Next, this text is provided as an input to a TTS (Automatic Speech Synthesis) model. The TTS engine now has the South Indian language text, for example Kannada, Telugu, Tamil and Malayalam, processes the data, applies appropriate pronunciation, and changes the text to grammatically correct speech. In this step, natural sounding audio is produced in the target South Indian language that is suitable for dubbing or playback.



```
2 files
transcription_Kannada.mp3(audio/mpeg) - 1633152 bytes, last modified: 9/26/2025 - 100% done
Do not Trust Anybody - Panchatantra In English - Moral Stories for Kids - Children's Fairy Tales.mp4(video/mp4) - 5919914
bytes, last modified: 9/26/2025 - 100% done
Saving transcription_Kannada.mp3 to transcription_Kannada (2).mp3
Saving Do not Trust Anybody  - Panchatantra In English - Moral Stories for Kids - Children's Fairy Tales.m
MoviePy - Writing audio in temp_original.wav
MoviePy - Done.
Detected speech start at 90 ms
Moviepy - Building video video_with_aligned_audio.mp4.
MoviePy - Writing audio in video_with_aligned_audioTEMP_MPY_wvf_snd.mp4
MoviePy - Done.
Moviepy - Writing video video_with_aligned_audio.mp4

t: 100%|██████████| 5280/5282 [01:00<00:00, 82.88it/s, now=None]WARNING:py.warnings:/usr/local/lib/python
  warnings.warn("Warning: in file %s, "%(self.filename)+

Moviepy - Done !
Moviepy - video ready video_with_aligned_audio.mp4
```

Figure 7. Final dubbed video output.

The proposed VoiceBridge framework is an advanced, low-cost, AI-backed system that can enable dubbing of multilingual video content into Indian regional languages effectively creating accessibility, cost-effective option, and linguistic fidelity. The structure is designed around the idea of combining multiple existing open-source libraries and models together into a modular pipeline to enable the translation and dubbing of English video to Indian regional languages like Kannada, Tamil, Telugu, and Malayalam. The implementation which is mostly in Python, utilizes ffmpeg-python to extract and merge audio-video, Flask to manage backend processing and web uploads, and Hugging Face Transformers for Text Translation (TT) based on IndicTrans2 and MarianMT, which both support Indian languages. VoiceBridge incorporates OpenAI Whisper and Vosk for Automatic Speech Recognition (ASR), allowing for accurate and reliable transcriptions for various accents of English. In the Text-to-Speech (TTS) part of the solution, Coqui TTS and Indic-TTS perform speech synthesis that sounds more like real human's speech in regional languages, and the Wav2Lip model can then be activated to provide more accurate lip-sync between the translated audio and video frames. The TTS system effectiveness was measured from common speech recognition metrics, Word Error Rate (WER) and Character Error Rate (CER), by measuring the performance of the transcriptions. The VoiceBridge produced the best overall results of all the models tested, with a WER of 11.9% and CER of 11.09%. Results were better in comparison to the W2V2-BERT (WER: 17%, CER: 21%), WhisperHindi (WER: 23.14%, CER: 14.0%) and Wav2Vec2 (WER: 30%, CER: 17%) models. These results validate that VoiceBridge produced improved transcription performance with the least number of transcription errors, high linguistic accuracy, and improved alignment for multilingual video dubbing.

Table 1. Performance comparison of ASR models (WER and CER)

Model	WER	CER
Wav2Vec2	30%	17%
DubWise	22.76%	36.53%
TeLeS	29.80%	12.86%
Whisper-Hindi	23.14%	14.0%
W2V2-BERT	17%	21%
VoiceBridge	11.9%	11.09%

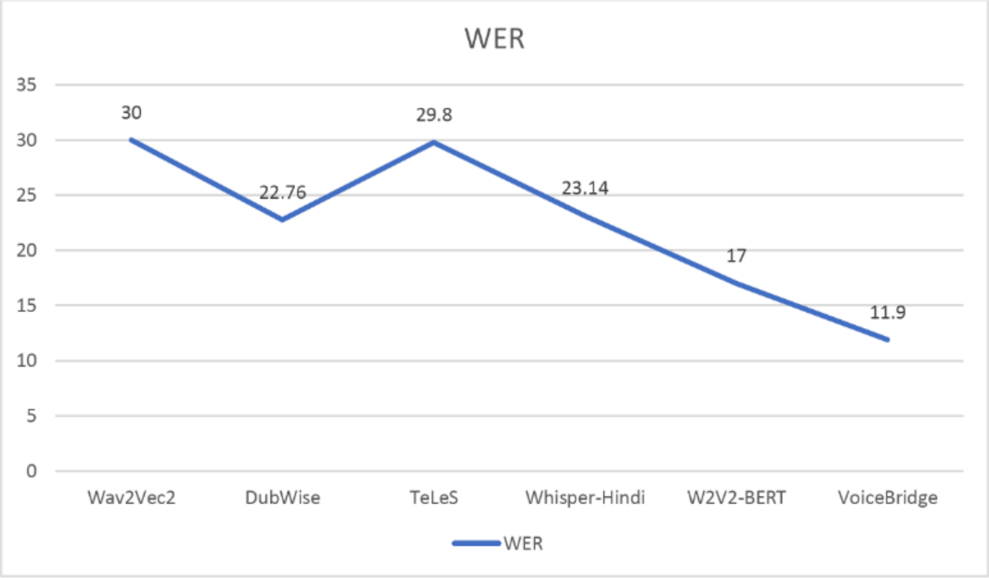


Figure 8. Comparative analysis of WER for different ASR models.

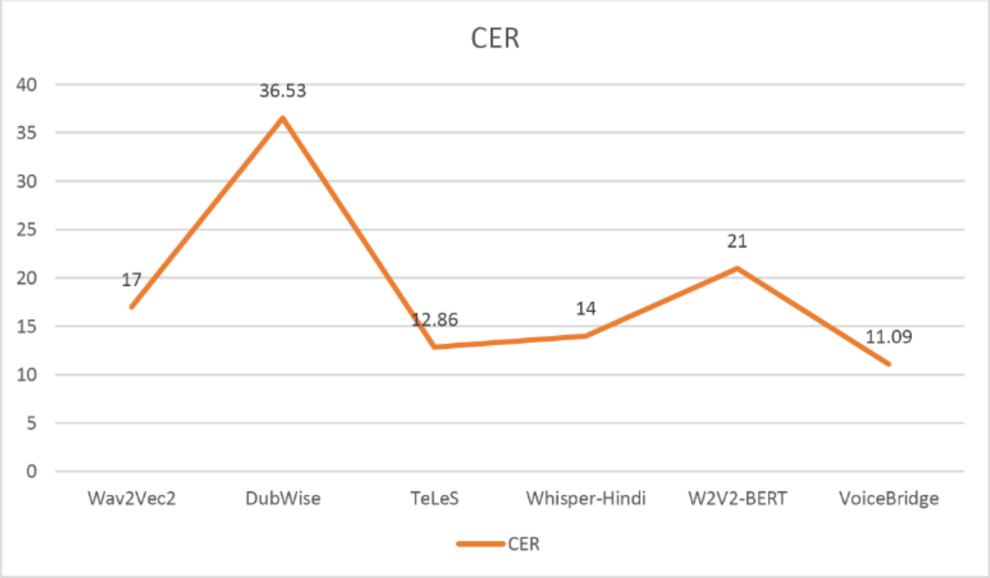


Figure 9. Comparative analysis of CER for different ASR models.

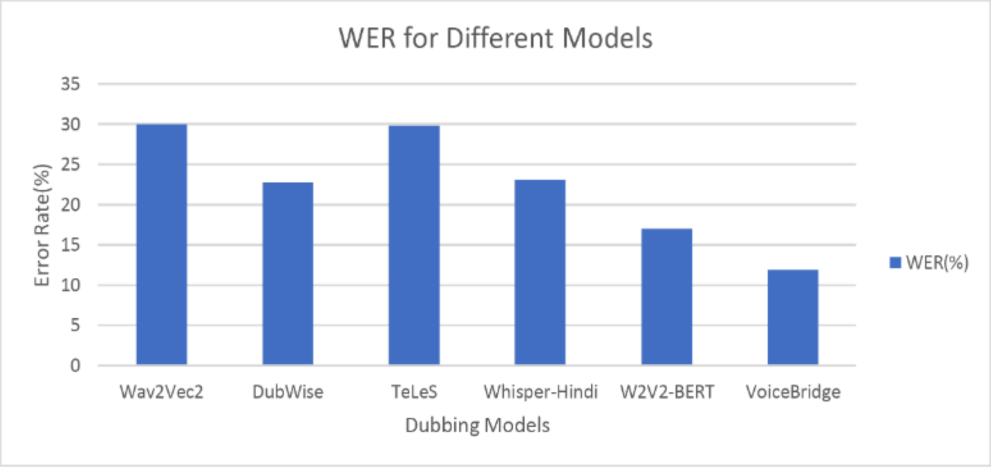


Figure 10. Comparative analysis of WER for different ASR models.

Ultimately, we embed the created South Indian language audio back into the video file, although proper lip-sync and timing accuracy is not achieved yet. The quality of the dubbing is actually evaluated based on the measures of Word Error Rate (WER) and Character Error Rate (CER). In this evaluation, the WER is 11.9% (12 or so word differences per 100), and CER is 11.09% (11 or so character differences per 100). The timing alignment is still a limitation, although these scores show robust accuracy in both speech recognition and translation accuracy with minimal issues with recognition of a few pronunciation errors or function words.

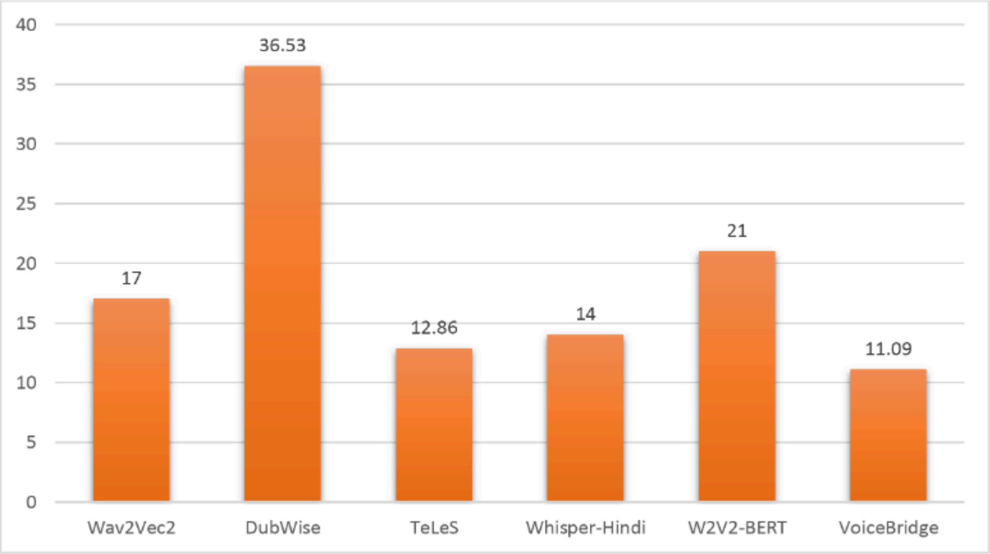


Figure 11. Comparative analysis of CER for different ASR models.

5. Conclusion

The VoiceBridge system addresses the significant challenge of language barriers in online video learning by offering an affordable, open-source, and culturally aware dubbing solution suitable for Indian regional languages. It uses an integrated pipeline of ASR, TT, and TTS technologies and successfully converts English videos into natural, regionally accurate audio outputs, it achieves promising WER and CER scores that indicate reliable and accurate transcription and translation quality. Although current limitations include a lack of precise lip-sync and a noticeable 90 millisecond audio delay, the results convey the framework’s potential for large-scale deployment in education, awareness campaigns, and rural outreach, by enabling cost-effective and accessible dubbing for low-resource languages. VoiceBridge helps connect people across the digital world while celebrating their languages and cultures.

References

[1] YouTube in education, Wikipedia, [online]. Available: https://en.wikipedia.org/wiki/YouTube_in_education.Accessed:2,2025.

[2] UNESCO, UNESCO survey highlights measures taken by countries to limit impact of COVID-19 school closures, UNESCO, 2023. [Online]. Available: https://www.unesco.org/en/articles/unesco-survey-highlights-measures-taken-countries-limit-impact-covid-19-school-closures.Accessed:Sep.2,2025.

[3] R. K. Nale, S. Bagal, H. Bhoite, S. Ghadge and S. Mohite, Text translation for English education videos into regional languages, International Research Journal of Modernization in Engineering Technology and Science 6(10) (2024). [Online]. Available: https://doi.org/10.56726/IRJMETS62629.

[4] Usage statistics of content languages for websites, Apr. 2022. [Online]. Available: https://w3techs.com/technologies/overview/content language.

[5] L. Moses, Bridging the digital language divide: policy and innovation, Digital Futures J. 7(2) (2023), 45-62.

[6] Census of India, C-17 population by bilingualism and trilingualism, 2011. [Online]. Available: https://web.archive.org/web/20191113211224/http://ww.censusindia.gov.in/2011census/C-17.html.

[7] H. Sheth, India’s active internet population likely to reach 900 million by 2025. Report, June 2021. [Online]. Available: https://www.thehindubusinessline.com/info-tech/indias-active-internet-population likely-to-reach-900-million-by-2025-report/article34714569.ece.

[8] V. Venkataraghavan, S. Sivapatham and A. Kar, Wav2Lip bridges communication gap: Automating lip sync and language translation for Indian languages, IEEE Access, 11 (2023), pp. xxxx-xxxx. doi: 10.1109/ACCESS.2023.xxxxx.

[9] A. Mahaganapathy and K. Sarveswaran, A survey and evaluation of text-to-speech system for the Tamil language, Natural Language Processing Journal 12 (2025), p. 100171. [Online]. Available: https://doi.org/10.1016/j.nlp.2025.100171.

[10] B. Meenakshi, M. W. Hussain and M. A. Sai, Real-time multilingual speech translation for peer communication, International Research Journal on Advanced Engineering Hub (IRJAEH), Vol. 2893 2025. [Online]. Available: https://www.researchgate.netpublication/393081843_Real-Time_Multilingual_Speech_Translation_for_Peer_Communication.

[11] V. V. Vijayabhaskarareddy, B. V. Venkata Prasad, B. Ramesh, G. Arvind and K. Rakesh, AI enhanced video language translation, IOSR Journal of Computer Engineering (IOSR-JCE) 27(1) (2025), 49-55. [Online]. Available: https://www.iosrjournals.org/iosr-jce/papers/Vol127-issue/Ser-2/G2701024955.pdf.

[12] S. K. Pulipaka, C. K. Kasaraneni, S. S. M. Kosaraju and V. N. S. Vemulapalli, Machine translation of english videos to indian regional languages using open innovation, International Journal of Computer Applications 175(1-5) (2019). [Online]. Available: https://www.researchgate.net/publication/338177583 Machine Translation of English Videos to Indian Regional Languages using Open Innovation.

[13] A. Dasare and K. T. Deepak, Performance assessment of voice conversion models using speech production-based parameters, Comput. Speech Lang. 95 (2025), 101853. [Online]. Available: https://doi.org/10.1016/j.csl.2025.101853.

[14] S. Bano, P. Jithendra, G. L. Niharika and S. Yalavarthi, Speech to Text Translation enabling Multilingualism, Proc. 2022 IEEE Int. Conf. Innov. Technol. (INOCON), Bengaluru, India, 2022, pp. 1-5. doi: 10.1109/INOCON50539.2022.9298280.

[15] R. Kannojia, A. K. Singh, I. Sharma and S. Gupta, Gen AI driven multilingual audio dubbing and synthesis system for cross-language video platforms, Bohrium, 2025. [Online]. Available: https://www.bohrium.com/paper-details/gen-ai-driven-multilingual-audio-dubbing-and-synthesis-system-for-cross-language-video-platforms/1152611458563964934-64194.

[16] R. Kannojia, A. K. Singh, I. Sharma and S. Gupta, Gen AI driven multilingual audio dubbing and synthesis system for cross language video platforms, ScienceDirect/Elsevier, 2025. [Online]. Available: https//www.sciencedirect.com/.

[17] X. Liu, M. Chen and Y. Zhao, TTS: Multi-modal text-to-speech of multi-scale style control for dubbing ScienceDirect/Elsevier, 2024. [Online]. Available: https://www.sciencedirect.com/.

[18] S. Kumar, L. Wang and D. Patel, Advancements in End-to-End Audio Style Transformation, MDPI, 20024. [Online]. Available: https://www.mdpi.com/.

[19] H. Zhang, P. Mehta and R. Srinivasan, Seeing the Sound: Multilingual Lip Sync for Real- Time Face Generation, MDPI, 2023/2024. [Online]. Available: https://www.mdpi.com/.

[20] J. Lee and K. Park, Audio-Driven Talking Face Generation with Stabilized Lip Movement, SpringerLink, 2024. [Online]. Available: https://link.springer.com/.

[21] V. Reddy, N. Sharma and A. Bose, Generating dynamic lip-syncing using target audio in a multimedia system, ScienceDirect, 2024. [Online]. Available: https://www.sciencedirect.com/.

[22] F. Zhao, T. Chen and W. Hu, Audio-visual speech synthesis using vision transformer-enhanced networks, SpringerLink, 2024. [Online]. Available: https://link.springer.com/.

[23] L. Singh, A. Roy and J. Kim, Speech driven video editing via an audio-conditioned diffusion model, ScienceDirect, 2024. [Online]. Available: https://www.sciencedirect.com/.

[24] D. Verma and S. Tripathi, Perceptual Evaluation of Audio-Visual Synchrony Grounded in Deep Learning, SpringerLink, 2024. [Online]. Available: https://link.springer.com/.

[25] P. Sharma, R. Gupta and N. Ahmed, Automatic Visual Lip Reading: A Comparative Review of Machine Learning Approaches, ScienceDirect, 2025. [Online]. Available: https://www.sciencedirect.com/.

[26] M. Gonzalez, E. Rodriguez and L. Perez, Evaluation of end-to-end continuous Spanish lipreading systems, SpringerLink, 2025. [Online]. Available: https://link.springer.com/.

[27] S. Deshmukh, R. Patel and K. Singh, Multilingual video dubbing - a technology review and current challenges, ResearchGate, 2023-2024. [Online]. Available: https://www.researchgate.net/.

[28] C. Wang, R. Li and M. Gomez, Exploring the Modalities of Audiovisual Translation: Focus on Cross-Language Synchrony, ResearchGate/ MDPI, 2024. [Online]. Available: https://www.mdpi.com/.

[29] A. Banerjee and L. Thomas, The Impacts of Video Dubbing on Non-English Major Students' Speaking Skills, ResearchGate, 2025. [Online]. Available: https://www.researchgate.net/.

[30] P. Mehta and R. Das, Video dubbing as a strategy for reducing foreign language speaking anxiety levels, ResearchGate, 2025. [Online]. Available: https://www.researchgate.net/.

[31] J. Kapoor and D. Lee, Deepfake Video Detection: Challenges and Opportunities, SpringerLink, 2024. [Online]. Available: https://link.springer.com/.