IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

# Building a Unified Multimedia Platform: Integrating Text-to-Speech Transcription, Audiobook Creation, Video Captioning, Audio Video Merging and Summary Generator for Enhanced Accessibility and User Experience.

Rushikesh Deshmukh
Department of Information Technology
Datta Meghe College of Engineering,
Navi Mumbai, India

Chetan Ahire
Department of Information Technology
Datta Meghe College of Engineering,
Navi Mumbai, India

Keshav Gadhari
Department of Information Technology
Datta Meghe College of Engineering,
Navi Mumbai, India

Siddhesh Bole
Department of Information Technology
Datta Meghe College of Engineering,
Navi Mumbai, India

Prof. Geeta Arwindekar
Department of Information Technology
Datta Meghe College of Engineering,
Navi Mumbai, India

***Abstract*:** The research explores the development of a unified multimedia system geared towards improving content access and creation by incorporating several major features such as Text-to- Speech, automated transcription, video captioning, audiobook creation, and audio video integration. The web application provides a comprehensive solution for users who face difficulties due to visual, auditory, or cognitive factors by transforming text into speech, generating captions for videos, and publishing audiobooks for text-to-speech listening. Finally, the users can combine audio and video material for more enhanced multimedia presentations.

**Keywords-** Multimedia Platform, Text-to-Speech, Transcription, Caption Generation, Audiobook Creation, Audio-Video Merging, Accessibility, Content Enhancement, Unified Platform.

## I. INTRODUCTION

We live in a digital world, and the production of multimedia content has greatly changed the way in which we communicate, learn and share information in the society. This project proposes a new platform that can help simplify the processing of multimedia content and at the same time improve its accessibility and usability for a large number of users. Some of the core features of this platform includes Subtitle Generation, Audio-video merger, Translation App, Text summary generator and Audiobook Making. Each of these features is targeted to solve certain problems in creation and use of multimedia contents, thus making the platform a useful tool in many applications. One of the major characteristics of this project is Text-to-Speech Transcription. This functionality employs the latest speech recognition technology to convert spoken audio into written text with a high level of accuracy. It captures meetings, lectures, interviews, or podcasts and eliminates the need for manual transcription, thus saving users a lot of time and energy. Another important feature is the Caption Generation tool that generates automatic subtitles for the videos. This functionality

makes sure that video content is easily accessible to people with hearing impairment and also helps non-native speakers to comprehend the information being conveyed.

The Live Translation (Talk App) feature is another great feature of this project. It provides real time translation of spoken words in course of a conversation and can help overcome linguistic barriers in international or multilingual settings. For media production, the Audio-Video Merger tool makes it easier to combine audio files with video. This feature comes in handy when adding voice overs, background music or sound effects to videos since it helps users to produce top notch and professional multimedia productions. In addition, the Audiobook Creation tool helps the user to easily convert textual information into an audio format. This feature can be particularly helpful for authors, publishers, and educators who are looking to convert their written content into an audio format. It also helps those who like to read or listen to materials, and those who are visually disabled, thus promoting and defending the right to learning for all.

## II. LITERATURE SURVEY

Sneha Tamboli et al. have given a thorough explanation of how TTS systems work, emphasizing how important they are for increasing accessibility and user engagement. The study examines various forms of voice recognition, classifying them according to speaker dependence, vocabulary range, and speech delivery techniques. The paper also explores how TTS systems are developed, highlighting the importance of Python libraries like Tkinter for graphical user interfaces (GUI) and other crucial tools that improve speech synthesis's usefulness and effectiveness. Aditya Ramani et al. has provided a thorough analysis of the various voice recognition engines and the numerous modules used in the automatic creation of video subtitles. The importance of integrating different Python-based modules, such as moviePy and pydub, which are essential for processing audio and video data, is highlighted in the study. A robust video editing library, moviePy enables smooth video clip manipulation, subtitle addition, and text-to-audio synchronization. However, Pydub plays a crucial role in managing audio processing duties such format conversion, audio file slicing, and audio file merging, which improves the precision and efficiency of subtitle generation.

## III. METHODOLOGY

The proposed Translive platform is designed as a unified multimedia solution, integrating multiple features to provide seamless multimedia processing. The system architecture consists of five main components:

### 1. AUDIOBOOK CREATION

The audiobook module enables text-to-speech conversion with an optional translation feature. Users can upload a .txt file, select a language, and hear the text spoken aloud. The system processes the text using a combination of JavaScript for file handling and speech synthesis, along with Flask for backend translation support.

PROCESS OVERVIEW

1. User Input: The user uploads a .txt file.
2. Preprocessing: JavaScript extracts the text from the file.
3. Translation: The extracted text is sent to the Flask backend, where the Google Translate API translates it into the selected language.
4. Text-to-Speech Conversion: The translated text is returned to the frontend and spoken aloud using the browser's SpeechSynthesis API.

**Output**: The user hears the audiobook in their chosen language.

### 2. CAPTION CRAFTING:

The Caption Crafting module takes a video as input, extracts its audio, converts speech to text, translates the text into Hindi, and overlays subtitles onto the video.

PROCESS OVERVIEW:

1. Upload and Validation:
   - The user uploads a video file through the web interface.
   - The system validates the file format.
2. Audio Extraction:
   - Audio is extracted from the video using moviepy.
3. Speech Recognition:
   - The extracted audio is converted to English text using SpeechRecognition.
4. Subtitle Generation:

- Subtitles are created using TextClip and positioned on the video using moviepy.

5. Video Rendering:
   - The subtitles are overlaid onto the video using Composite Video Clip
6. Download:
   - The user can download the processed video with subtitles.

**Output**: Video with subtitles (.mp4)

### 3. AUDIO-VIDEO SYNCHRONIZATION MODULE

The Audio-Video Synchronization module integrates an external audio file with a video by replacing the original audio applications, or content localization.

Process Overview :

1. Upload and Validation:
   - The user uploads both video and audio files using the HTML form.
   - The system verifies the file types and ensures both files are valid.
2. Audio Extraction and Replacement:
   - VideoFileClip from moviepy reads the video file.
   - AudioFileClip extracts the audio from the audio file.
   - The extracted audio is set as the new audio for the video using set_audio().
3. Rendering**:**
   - The final video with the new audio track is exported usingwrite_videofile().
   - libx264 codec is used for video compression.
4. Download**:**
   - The processed video is available for download via a generated link.

**Output:** Video with synchronized audio (.mp4)

### 4. VIDEO SUMMARY GENERATOR

The Video Summary Generation module extracts audio from a video, transcribes it to text using speech recognition, applies punctuation, and then generates a summarized version using an extractive text summarization algorithm.

PROCESS OVERVEW:

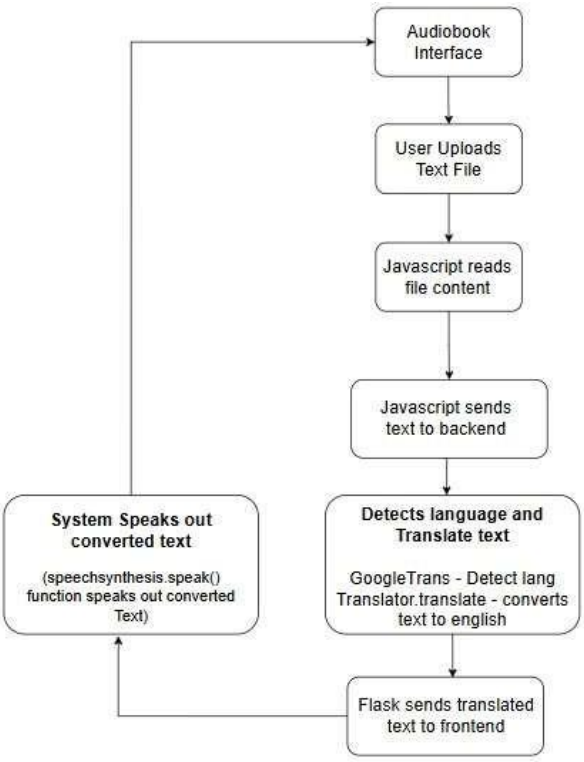
1. Audio Extraction:
   - VideoFileClip from moviepy extracts audio from the video file.
   - The audio is temporarily stored in .mp3 format using write_audiofile().
   - pydub converts this .mp3 to .wav for better compatibility with the speech recognition module.
2. Speech-to-Text Conversion:
   - SpeechRecognition library processes the .wav file using recognize_google() API.
   - The speech data is converted into raw textual format.
3. Punctuation Restoration:
   - DeepMultilingualPunctuation is used to restore punctuation to the transcribed text for readability and coherence.
4. Text Summarization:
   - LsaSummarizer (Latent Semantic Analysis) from sumy library performs extractive summarization.
   - A specified number of sentences (default 5) are selected to generate the final summary.
5. Output Generation:
   - The generated summary is saved as a .txt file in the output folder.

**Output:** Summarized Text File (summary.txt)

## IV. FLOWCHARTS

The following are the flowcharts of the different features in our project which consist of technical description.

### 1. AUDIOBOOK CREATION

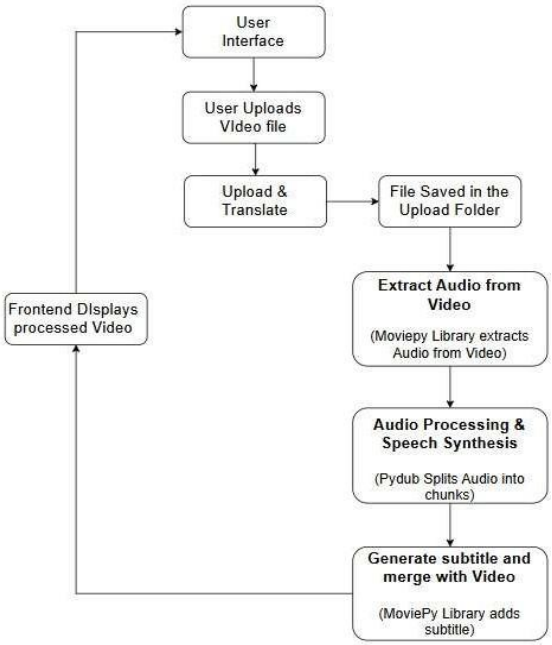



The audiobook workflow starts with the Audiobook Interface, where users upload a text file. JavaScript reads the file content and sends the text to the backend. The backend uses GoogleTrans to detect the language and translates the text into English if needed. The Flask backend then sends the translated text to the frontend. Finally, the system converts the text to speech using the speechSynthesis.speak() function, allowing users to listen to the converted text.

#### WORKFLOW STEPS:

1. User Uploads Text File – The audiobook interface allows users to upload a file.
2. JavaScript Reads & Sends Text – JavaScript processes the file and sends text to the backend.
3. Language Detection & Translation – GoogleTrans detects the language and translates it into English.
4. Flask Sends Translated Text – The backend sends the translated text to the frontend.
5. Text-to-Speech Conversion – The system uses speechSynthesis.speak() to read the text aloud.

## 2. CAPTION CRAFTING



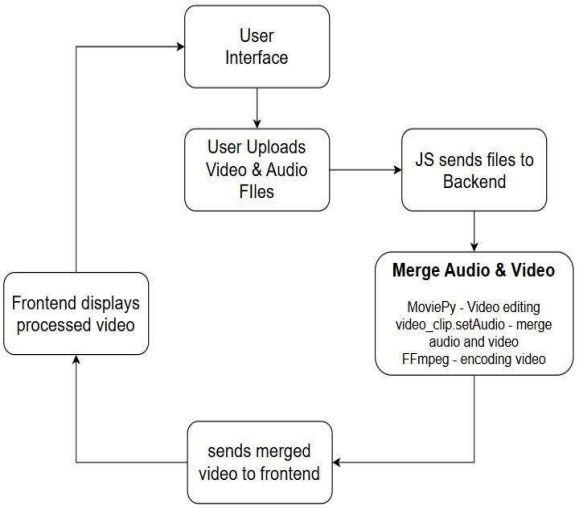

The caption crafting workflow begins with the User Interface, where users upload a video and set the number of sentences for the summary. The Backend Flask receives the file and initiates processing. The system first extracts audio using MoviePy, then Pydub splits the audio, and speech recognition converts it into text. A punctuation model is applied for readability, followed by summarization using Summy (LexRank & LSA) to generate a concise summary. The system displays processing progress to the user, and once completed, the final output is displayed, with an option to download the summary.

### WORKFLOW STEPS:

1. User Uploads Video & Sets Summary Length – Users interact via the UI.
2. Backend Processing – Flask receives the file.
3. Audio Extraction – MoviePy extracts audio from the video.
4. Speech Processing – PyDub splits audio, and speech recognition converts it to text.
5. Apply Punctuation – A punctuation model improves text readability.
6. Summarization – Summy (LexRank & LSA) generates a concise summary.

Progress Display & Output – Users see processing progress and can download the summary

## 3. AUDIO-VIDEO SYNCHRONIZATION MODULE



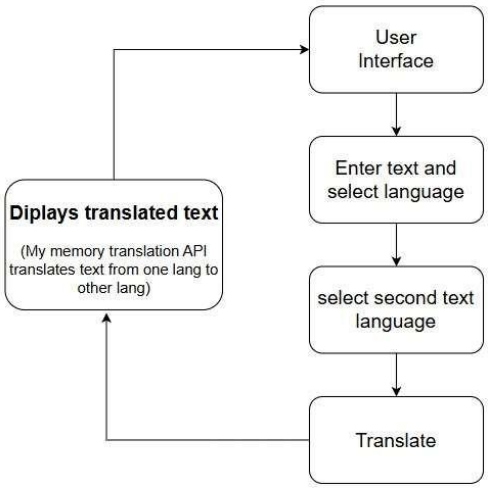

The audio-video synchronization process begins with the User Interface, where users upload video and audio files. JavaScript sends these files to the backend, where MoviePy merges the audio with the video

using video_clip.setAudio(). FFmpeg then encodes the synchronized video. The processed video is sent back to the frontend, where it is displayed as the final output.

**Workflow Steps:**

1. User Uploads Video & Audio – Users upload files via the interface.
2. JavaScript Sends Files to Backend – The files are sent for processing.
3. Merge Audio & Video – MoviePy merges the audio and video.
4. Encoding – FFmpeg encodes the synchronized video.
5. Processed Video Sent to Frontend – The final video is displayed to the user.

## 4. TEXT TRANSLATION

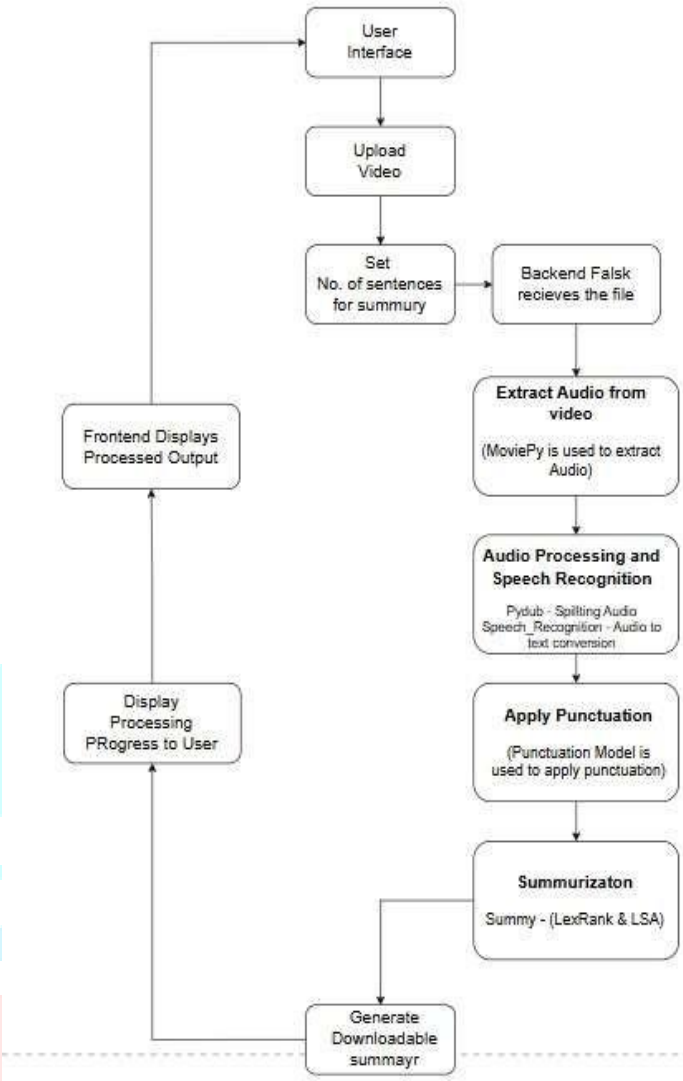



The text translation workflow for the audiobook begins with the User Interface, where users enter text and select the source language. They then choose the target language for translation. The system processes the request using My Memory Translation API, which translates the text from the selected source language to the target language. Finally, the translated text is displayed to the user.

**Workflow Steps:**

1. User Interface – Users enter text and select the source language.
2. Select Target Language – Users choose the language into which the text should be translated.
3. Translation Process – The system processes the translation using My Memory Translation API.
4. Display Translated Text – The translated text is shown to the user.

## 5. SUMMARY GENERATOR




The video summary generator starts with the User Interface, where users upload a video and set the number of sentences for the summary. The Backend Flask receives the file and extracts the audio using MoviePy. The extracted audio undergoes processing and speech recognition with PyDub, converting it into text. A punctuation model is applied to enhance readability, followed by summarization using Summy (LexRank & LSA). The system displays processing progress, and once completed, the final summary is generated and made available for download.

**Key Workflow Steps:**

1. User Uploads Video & Sets Summary Length – Users interact via the UI.
2. Backend Processing – Flask receives the video file.
3. Audio Extraction – MoviePy extracts audio from the video.
4. Speech Processing – PyDub splits audio, and speech recognition converts it to text.
5. Apply Punctuation – A punctuation model improves text . readability
6. Summarization – Summy (LexRank & LSA) generates a concise summary.
7. Progress Display & Output – Users see processing progress . and can download the summary.

## V. RESULTS AND DISCUSSION

### Performance Metrics and Evaluation

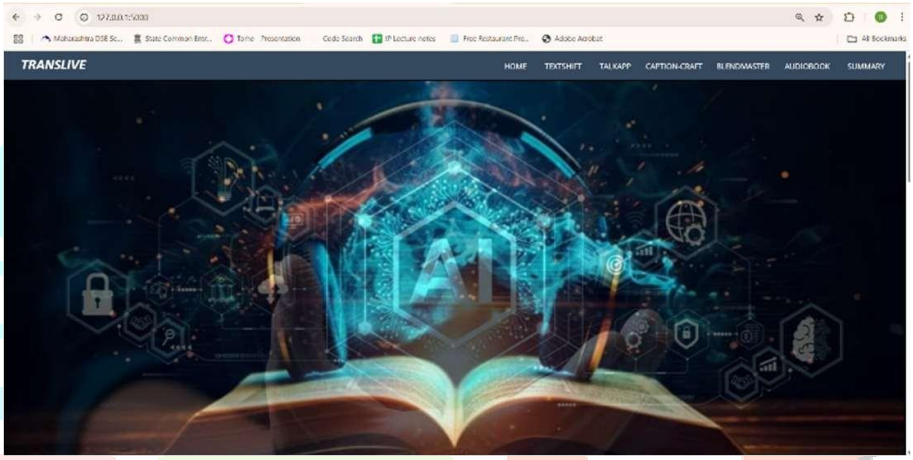
The system's performance was assessed using standard evaluation metrics, including Confusion matrix, speech recognization accuracy, Character Error Rate (CER), ROUGE (Recall-Oriented Understudy for Gisting Evaluation) and Word Error Rate (WER) to evaluate the quality of text summarization and machine translation models.

| Metrics | Results |
|---|---|
| Character Error Rate | 0.06 |
| Rougue Score | 0.66 |
| Word Error Rate | 0.27 |

Table 1. Evaluation metrics for system.

The below are the glimpse of user interface where user will interact

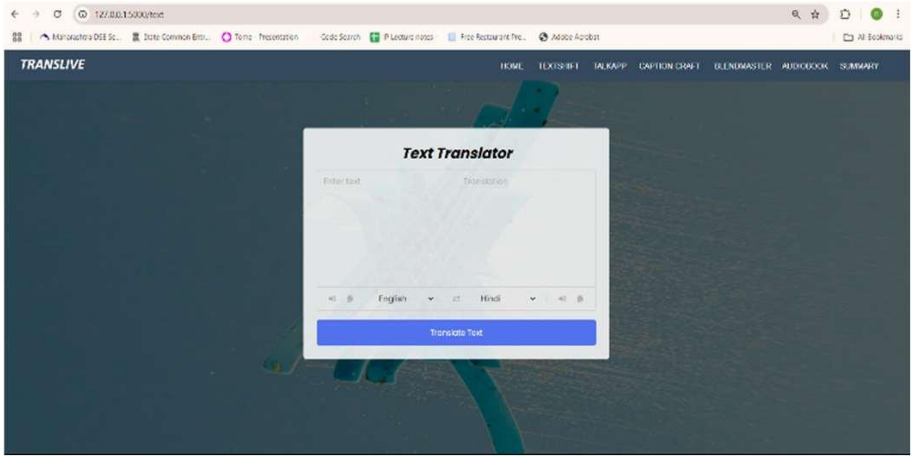
1. **Home Page**




On the Translive Home Page, you'll find easy-to-use options to transform your media. Simply choose what you want to do:

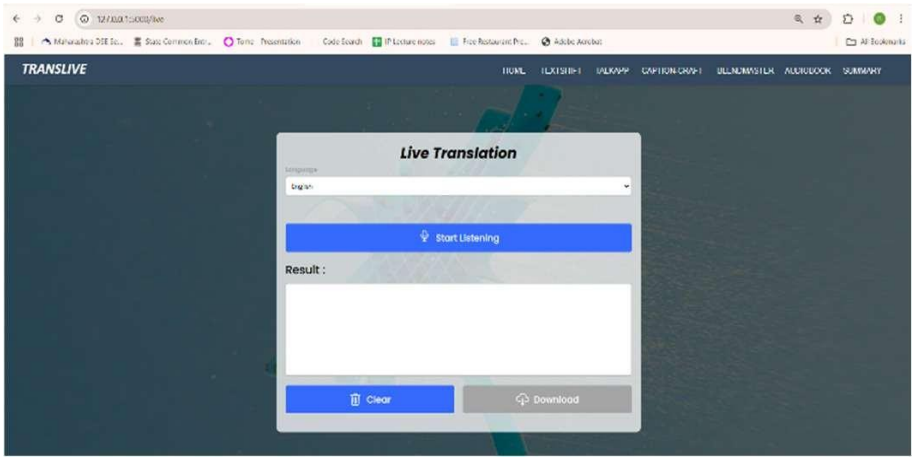
- **Text-to-Speech Transcription:** Convert written text into clear, natural-sounding speech.
- **Audiobook Creation:** Turn your text into an engaging audiobook.
- **Video Captioning:** Generate captions for your videos quickly and accurately.
- **Audio-Video Merging:** Sync audio with videos for perfect alignment.
- **Video Text Summary Generator:** Get a short, clear summary of any video.

2. **Text Translator**



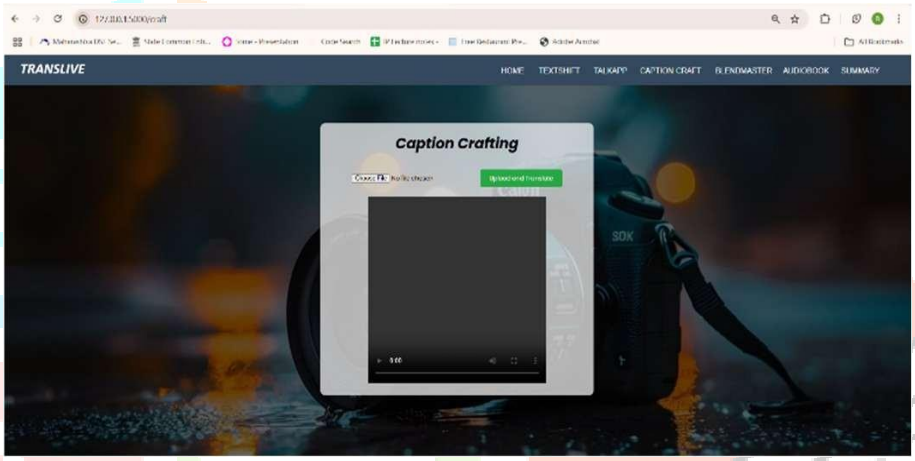

The Text translator page one can easily translate your text into multiple languages with Translive. just enter your text, select your desired language, and get an accurate translation in seconds.

### 3. Live Translation



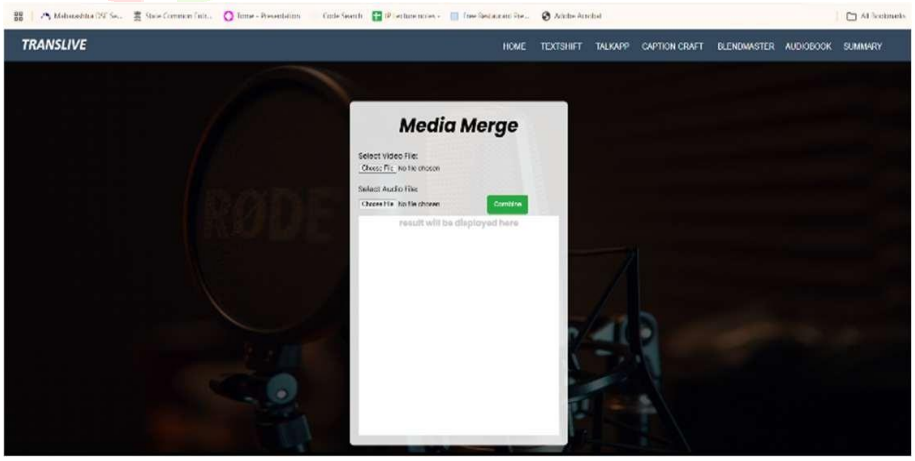

The Live translation page translate your speech in real-time and provide output in the form of text. This feature helps to communicate easily across different languages without any barriers.

### 4. Caption Crafting



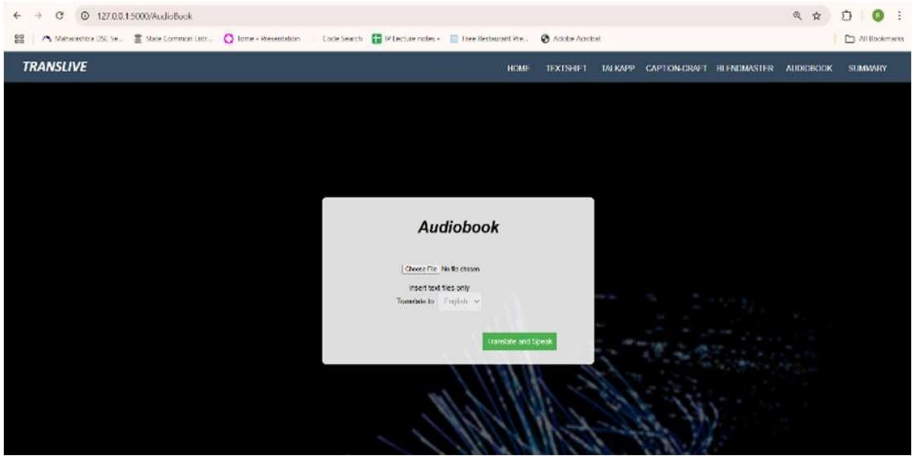

The feature generates accurate and engaging captions for your videos. It is perfect for making your content accessible and enhancing viewer experience.

### 5. Blend Master



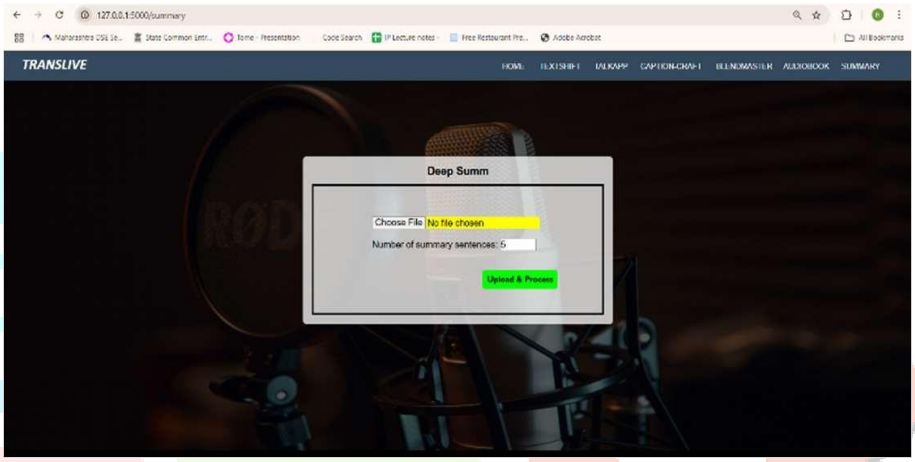

The feature effortlessly sync audio with video for smooth and accurate playback. It generates brief summary for the video.

### 6. Audiobook




The audiobook generator converts your text into clear, natural- sounding audio.

### 7. Video Text Summary Generator




The Video Text Summary Generator quickly analyzes your video content and provides a concise text summary. This feature is ideal for professionals, students, and content creators who need quick insights from webinars, lectures, or meeting.

## VI. CONCLUSION

The Translive is an important step in creating a more accessible and efficient multimedia content structure. The key features of the system include Text to Speech Transcription, Audiobook Creation, Video Captioning, Audio-Video Synchronization, and the Video Text Summary Generator. The platform offers the Text to Speech Transcription feature, which saves time by turning text into naturally sounding speech — a perfect solution for visually impaired users and audiobook. The Audiobook creation module builds on this functionality, adding a convenient format for listening to content on the go. And specifically, the Video Captioning tool improves video accessibility for hearing-impaired individuals and also aids understanding for those who do not speak the language natively by producing precise subtitles. By automating the process of synchronizing external audio with video, the Audio-Video Synchronization module streamlines the workflow of content creators, educators, and media producers who often rely on both audio and video elements in their work. On the other hand, the Video Text Summary Generator employs AI algorithms to summarize. On the other hand, the Video Text Summary Generator, a cutting-edge tool powered by advanced AI algorithms, summarizes long video content into concise and informative text, helping users save time and effort. In summary, translive is a complete and cutting-edge multimedia solution that successfully fills in the gaps in multimedia management and content accessibility. It makes the digital world more connected and inclusive by enabling users to easily produce, manage, and consume multimedia content.

## VII. REFERENCES

**[1]** Sneha Tamboli, Pratiksha Raut, Lavkush Sategaonkar, Anjali Atram, Shubham Kawane, Prof. V.K. Barbudhe., "A Review Paper On Text-To-Speech Convertor". ICOET, Yavatmal. 2022.

**[2]** Aditya Ramani, Asmita Rao, Vidya V, VR Badri Prasad., "Automatic Subtitle Generation for videos". ICACCS, Bangalore. 2020.

**[3]** Vinnarasu A., Deepa V. Jose., "A Review Paper On Speech to text conversion and summarization for effective understanding and documentation".Bengaluru, Karnataka, India. 2019

**[4]** Kaveri Kamble, Ramesh Kagalkar., "A Review Paper On Translation of Text to Speech Conversion for Hindi Language". Volume 3 Issue 11, November 2014

**[5]** Dr. M. Anusha, K Pavan Kumar, Srikanth Vemuri, V. Madhusudhana Reddy, T. Vaishnav., "A Review Paper On Speech-to-Text and Text-to-Speech Recognition".Guntur, Andhra Pradesh, India -522302

**[6]** D.Sasirekha, E.Chandra," Text to Speech: A Simple Tutorial", International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-1, March 2012.

**[7]** S. Venkateswarlu, D. B. K. Kamesh, J. K. R. Sastry, Radhika Rani. "Text to Speech Conversion." Indian Journal of Science and Technology, vol. 9, no. 38, 2016.

**[8]** Md. Jalal Uddin Chowdhury, Ashab Hussan. "A review-based study on different Text-to-Speech technologies." arXiv preprint arXiv:2312.11563, 2023.

**[9]** Mohammad Reza Hasanabadi. "An overview of text-to-speech systems and media applications." arXiv preprint arXiv:2310.14301, 2023.

**[10]** Swaroopa Shastri, Shashank Vishwakarma. "An Efficient Approach for Text- to-Speech Conversion Using Machine Learning and Image Processing Technique." International Journal of Engineering and Manufacturing, vol. 13, no. 4, 2023.

**[11]** Xinhao Mei, Xubo Liu, Mark D. Plumbley. "Automated Audio Captioning: An Overview of Recent Progress and New Challenges." EURASIP Journal on Audio, Speech, and Music Processing, 2022.

**[12]** IRJET Journal, Kanna Velusamy. "Automatic Susbtitle Generation for Sound in Videos." International Research Journal of Engineering and Technology, 2016.

**[13]** Polepaka Sanjeeva, Vanipenta Balasri Nitin Reddy, Jagirdar Indraj Goud, Aavula Guru Prasad. "TEXT2AV – Automated Text to Audio and Video Conversion." E3S Web of Conferences, vol. 430, 2023.

**[14]** Xinhao Mei, Xubo Liu, Mark D. Plumbley. "A Comprehensive Survey of Automated Audio Captioning." arXiv preprint arXiv:2201.00146, 2022.

**[15]** Yi Ren, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, Tie-Yan Liu. "Almost Unsupervised Text to Speech and Automatic Speech Recognition." arXiv preprint arXiv:1905.06791, 2019.

**[16]** Xun Gong, Yu Wu, Jinyu Li, Shujie Liu, Rui Zhao, Xie Chen, Yanmin Qian. "Advanced Long-Content Speech Recognition With Factorized Neural Transducer." arXiv preprint arXiv:2403.13423, 2024.

**[17]** Md. Jalal Uddin Chowdhury, Ashab Hussan. "A Review-Based Study on Different Text-to-Speech Technologies." arXiv preprint arXiv:2312.11563, 2023.

**[18]** Xinhao Mei, Xubo Liu, Mark D. Plumbley. "Automated Audio Captioning: An Overview of Recent Progress and New Challenges." EURASIP Journal on Audio, Speech, and Music Processing, 2022.

**[19]** Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, Gunhee Kim. "AudioCaps: Generating Captions for Audios in The Wild." Proceedings of NAACL-HLT 2019, 2019.