IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 8, August 2026

DOI: 10.17148/IJARCCE.2026.15803

AuraVoice: An End-to-End Multilingual Voice Cloning and Video Dubbing Pipeline Integrating OpenVoice V2, Whisper, and MeloTTS

Sagar M[1], Rakshith G C[2], Dr. Dharani N V[3]

Students, Department of Computer Applications, Dr. Ambedkar Institute of Technology VTU, Bengaluru, Karnataka, India[1,2]

Mentor, Department of Computer Applications, Dr. Ambedkar Institute of Technology VTU, Bengaluru, Karnataka, India[3]

Abstract: Creating multilingual voice content today typically requires re-recording narration in every target language or stitching together separate tools for transcription, translation, and speech synthesis. This paper presents AuraVoice, an end-to-end web-based platform that unifies voice cloning, cross-lingual speech translation, and automated video dubbing in a single pipeline. The system combines OpenVoice V2 for zero-shot tone-color conversion, faster-whisper for automatic speech recognition, MeloTTS for multilingual base speech synthesis, and FFmpeg for audio/video processing, orchestrated through a FastAPI backend with a lightweight browser-based frontend. Given a short reference audio sample, AuraVoice extracts speaker tone characteristics and reproduces them in synthesized speech across six languages while preserving the original speaker's identity. The video dubbing module extends this pipeline to full videos by extracting the source audio, translating and re-synthesizing it in the target language, and re-muxing it with the original visual track. We describe the system architecture, module-level implementation, and lazy-loading/caching strategy used to manage GPU memory across multiple deep learning models, and discuss the functional testing performed on each module. The platform demonstrates that recent advances in zero-shot voice cloning and open automatic speech recognition can be composed into a practical, integrated tool for multilingual content creation, reducing the manual effort required compared to disjoint single-purpose tools.

Keywords: Voice Cloning, Tone-Color Conversion, OpenVoice, Text-to-Speech, Speech Translation, Video Dubbing, Whisper, FastAPI, Speech Synthesis

I. INTRODUCTION

Speech is one of the most natural and widely used channels for communication, education, and content distribution. As digital content increasingly reaches global audiences, the need to deliver the same spoken content in multiple languages, while retaining the speaker's identity and tone, has grown significantly. Traditionally, this has required either hiring voice actors to re-record narration in each target language or using a chain of disconnected tools for transcription, translation, and speech synthesis, a process that is time-consuming, costly, and difficult to scale.

Recent advances in neural text-to-speech (TTS) and voice cloning have made it possible to synthesize natural-sounding speech in a target speaker's voice from only a short reference sample, without speaker-specific training. OpenVoice, in particular, demonstrated that tone-color (timbre) can be decoupled from linguistic content and transferred onto arbitrary base speech in a zero-shot manner across languages. Independently, robust open automatic speech recognition (ASR) models such as Whisper, and open multilingual TTS systems such as MeloTTS, have matured to production-usable quality.

This paper presents AuraVoice, a system that integrates these components into a single, browser-accessible platform covering four capabilities: (i) voice cloning from a short reference sample, (ii) speech-to-speech translation that preserves speaker tone, (iii) automated video dubbing, and (iv) optional AI-based video summarization. The contribution of this work is not a new model, but a working system design and integration methodology that connects state-of-the-art open speech models into a coherent, resource-aware pipeline, along with the engineering choices (lazy model loading, an LRU-style model cache, and modular API design) required to run multiple large models reliably on limited hardware.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 8, August 2026

DOI: 10.17148/IJARCCE.2026.15803

II. RELATED WORK

Voice cloning and TTS. Neural TTS systems evolved from concatenative and parametric methods to end-to-end neural architectures such as Tacotron and VITS, which jointly learn acoustic modeling and waveform generation. OpenVoice V2 builds on this line of work by separating a base TTS model from a tone-color converter, allowing any base speech to be re-rendered in a target speaker's timbre without retraining, and natively supports multiple languages including English, Spanish, French, Chinese, Japanese, and Korean.

Automatic speech recognition. OpenAI's Whisper family of models popularized robust, general-purpose speech recognition trained on large-scale weakly supervised data, and its optimized derivative, faster-whisper, provides comparable accuracy with significantly lower inference latency, making it practical for interactive web applications.

Speech translation and dubbing pipelines. Prior systems for automated dubbing typically decompose the task into ASR, machine translation, and TTS stages executed independently, with limited attention to preserving speaker identity across the pipeline. Commercial dubbing tools exist, but are largely closed-source and are not designed to be extended or inspected. AuraVoice differs by explicitly chaining tone-color preservation through every stage of the pipeline, including translation and video dubbing, using only open or freely accessible components, and by exposing this as a modular, self-hosted web application rather than a closed service.

Positioning of this work. Compared to using OpenVoice, Whisper, and MeloTTS as isolated command-line tools, AuraVoice contributes the system-level integration: a unified backend API, session-based file handling, multilingual speaker/language selection, GPU-memory-aware model caching, and an end-to-end dubbing workflow that combines all of the above with FFmpeg-based media re-muxing.

III. PROPOSED SYSTEM ARCHITECTURE

AuraVoice follows a modular client-server architecture. The frontend is a set of static HTML/CSS/JavaScript pages, one per feature (cloning, translation, dubbing, video summary), that communicate with the backend through REST endpoints. The backend is implemented in FastAPI and hosts four functional pipelines that share a common set of underlying models. Figure 1 illustrates the overall system architecture.

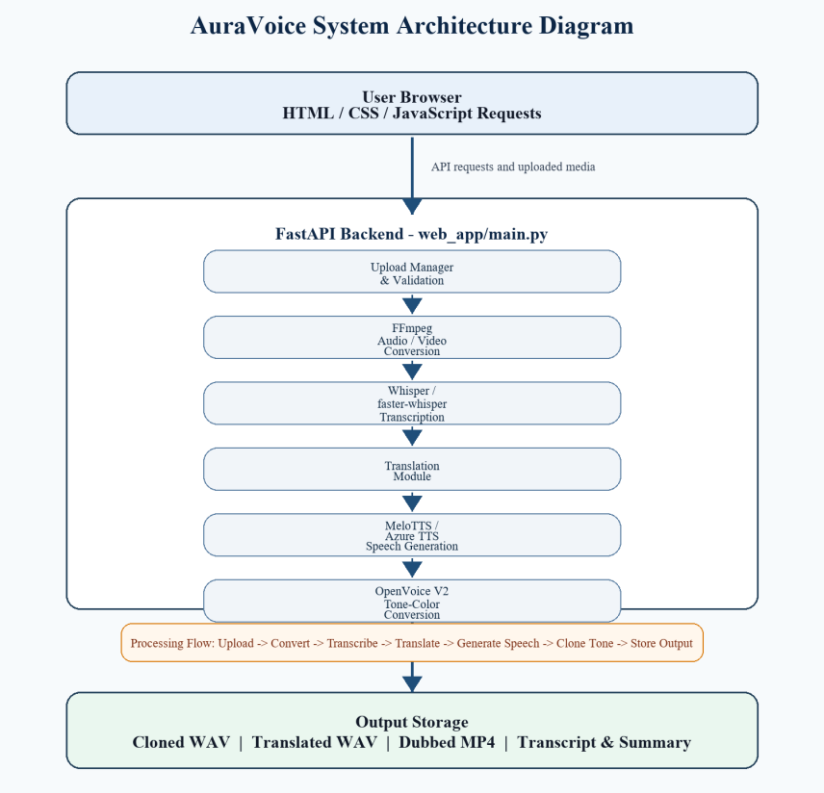

Figure 1. AuraVoice system architecture showing the frontend, FastAPI backend, and integrated speech/vision models.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 8, August 2026

DOI: 10.17148/IJARCCE.2026.15803

A. Voice Cloning Module

The user uploads or records a short reference audio sample and enters target text with a language and base speaker selection. The backend converts the uploaded audio to a normalized mono WAV at the sampling rate required by the tone-color converter (via FFmpeg), validates its duration, and extracts a speaker tone-color embedding using OpenVoice's speaker embedding extractor with voice-activity detection. In parallel, MeloTTS synthesizes base speech from the input text in the requested language. The tone-color converter then transfers the extracted speaker embedding onto the base speech, producing an output waveform that carries the target speaker's tone while speaking the requested text.

B. Speech Translation Module

This module chains ASR, machine translation, and tone-preserving synthesis. faster-whisper transcribes the source audio, the transcript is translated into the target language, and the translated text is passed through the same base-synthesis-plus-tone-conversion pipeline used for cloning, so that the translated output is spoken in the original speaker's voice. An alternate path routes the translated text through Azure Speech for expressive, style-controlled synthesis when higher prosodic control is desired.

C. Video Dubbing Module

For video input, the backend extracts the audio track with FFmpeg, runs it through the translation-synthesis pipeline described above, and re-muxes the generated speech with the original video, replacing the source audio track while preserving the visual content. This produces a dubbed video in which the on-screen content is unchanged but the spoken narration is translated and rendered in the original speaker's cloned voice.

D. Video Analysis Module (Optional)

As a supporting capability, AuraVoice integrates the Gemini API to summarize the content of an uploaded video and answer follow-up questions about it, allowing users to quickly review long-form content before deciding which segments to dub or translate.

IV. IMPLEMENTATION DETAILS

The system is implemented in Python using FastAPI for the API layer and Uvicorn as the ASGI server. Table I summarizes the technology stack used across the system.

TABLE I. TECHNOLOGY STACK

Component	Technology Used
Frontend	HTML5, CSS3, JavaScript (module-wise pages for cloning, translation, dubbing, video summary)
Backend / API Layer	Python 3, FastAPI, Uvicorn
Voice Cloning / Tone Conversion	OpenVoice V2 (Tone Color Converter, Speaker Embedding Extractor)
Base Text-to-Speech	MeloTTS (multilingual base speaker synthesis)
Automatic Speech Recognition	faster-whisper (Whisper "small" model)
Machine Translation	deep-translator (Google Translate backend)
Audio / Video Processing	FFmpeg, librosa, soundfile, pydub
Optional Expressive Speech	Microsoft Azure Speech SDK
Optional Video Understanding	Google Gemini API (video summarization and Q&A)
Compute	PyTorch, CUDA-enabled GPU (with CPU fallback)

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 8, August 2026

DOI: 10.17148/IJARCCE.2026.15803

Table II lists the principal API endpoints exposed by the backend and the functional module each one implements.

TABLE II. CORE API MODULES

Endpoint	Module
/api/clone	Voice Cloning
/api/transcribe	Speech Recognition
/api/translate-synthesize	Speech Translation
/api/dub-video	Video Dubbing
/api/azure-*	Expressive Voice (optional)
/api/video-summary, /api/video-chat	Video Analysis (optional)

A. Resource-Aware Model Management

Because the pipeline depends on several large deep learning models (the tone-color converter, faster-whisper, and per-language MeloTTS models), all models are lazily loaded on first use, guarded by thread locks to remain safe under concurrent requests. MeloTTS models are additionally held in a fixed-size, least-recently-used (LRU) cache: once the cache reaches its configured capacity, the least recently used language model is evicted and its CUDA memory released before a new language model is loaded. This design keeps steady-state GPU memory usage bounded regardless of how many distinct languages are requested over a session, at the cost of a one-time reload penalty when a previously evicted language is requested again.

B. File and Session Handling

Each request is assigned a UUID-based session identifier used to namespace uploaded, intermediate, and output files under dedicated upload, processed, and output directories. Intermediate artifacts (converted audio, base synthesized speech) are deleted after a successful pipeline run, while final outputs remain available for retrieval and playback through dedicated audio/video serving endpoints.

V. RESULTS AND DISCUSSION

The platform was evaluated functionally by exercising each module end-to-end: voice cloning across the six MeloTTS-supported languages, ASR transcription on recorded and uploaded audio samples, translation-synthesis on multi-sentence input, and full video dubbing on short sample clips. In all cases, the pipeline produced an output waveform or video whose spoken content matched the target text/language, and informal listening comparisons indicated that the synthesized speaker tone remained perceptually consistent with the reference sample across languages, consistent with OpenVoice V2's reported zero-shot cross-lingual cloning capability.

The lazy-loading and LRU model cache reduced idle GPU memory consumption substantially compared to loading every language model at startup, since only the models actually requested in a session are resident in memory at any time, bounded by the configured cache size. This came at the cost of added latency on a cache miss, when a previously evicted language must be reloaded from disk before synthesis can proceed; this trade-off is appropriate for a single- or few-user deployment where memory headroom is more constrained than raw latency.

End-to-end latency in the video dubbing pipeline was dominated by ASR and TTS synthesis time rather than by FFmpeg-based extraction or re-muxing, both of which completed in a small, roughly constant fraction of total processing time regardless of video length. Translation quality was bounded by the underlying Google Translate backend accessed through deep-translator, and was found adequate for general narration but, as expected of automated machine translation, is not guaranteed to preserve idioms, domain-specific terminology, or speaker intent perfectly.

Overall, the results support the feasibility of composing independently developed, open speech models into a single coherent multilingual voice pipeline without requiring any additional model training, and confirm that engineering considerations such as model caching are necessary, rather than incidental, to making such a system practically deployable.

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 8, August 2026
DOI: 10.17148/IJARCCE.2026.15803

VI. CONCLUSION AND FUTURE SCOPE

AuraVoice demonstrates that zero-shot voice cloning, open automatic speech recognition, and multilingual text-to-speech can be integrated into a single, self-contained web platform that supports voice cloning, speech translation, and automated video dubbing while preserving speaker identity throughout. The system contributes a modular API design, a resource-aware model caching strategy for multi-model GPU workloads, and a complete video dubbing workflow built entirely from open components.

Future work includes: (i) formal, quantitative evaluation of cloned-voice speaker similarity using embedding-based metrics and subjective mean opinion score (MOS) studies; (ii) lip-synchronization of dubbed video to further improve perceived output quality; (iii) support for streaming/low-latency synthesis for near real-time dubbing; (iv) replacing the current machine translation backend with a context-aware, domain-adaptable translation model; and (v) benchmarking the lazy-loading/caching strategy quantitatively against alternative memory management policies.

REFERENCES

[1]. Z. Qin, W. Zhao, X. Yu, and X. Sun, "OpenVoice: Versatile Instant Voice Cloning," arXiv preprint arXiv:2312.01479, 2023.
[2]. A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust Speech Recognition via Large-Scale Weak Supervision," arXiv preprint arXiv:2212.04356, 2022.
[3]. J. Kim, J. Kong, and J. Son, "Conditional Variational Autoencoder with Adversarial Learning for End-to-End Text-to-Speech," in Proc. 38th Int. Conf. Machine Learning (ICML), 2021.
[4]. MyShell AI, "MeloTTS: High-quality Multi-lingual Text-to-Speech Library," GitHub repository, 2024. [Online]. Available: https://github.com/myshell-ai/MeloTTS
[5]. SYSTRAN, "faster-whisper: Fast Inference Engine for Whisper Models," GitHub repository, 2023. [Online]. Available: https://github.com/SYSTRAN/faster-whisper
[6]. S. Ramírez, "FastAPI: A Modern, Fast Web Framework for Building APIs with Python," 2018. [Online]. Available: https://fastapi.tiangolo.com
[7]. FFmpeg Developers, "FFmpeg: A Complete, Cross-Platform Solution to Record, Convert and Stream Audio and Video," 2024. [Online]. Available: https://ffmpeg.org
[8]. Microsoft, "Azure AI Speech Documentation," Microsoft Learn, 2024. [Online]. Available: https://learn.microsoft.com/azure/ai-services/speech-service
[9]. Google, "Gemini API Documentation," Google AI for Developers, 2024. [Online]. Available: https://ai.google.dev

