IJARETY

International Journal of Advanced Research in Education and TechnologY (IJARETY)

Volume 12, Issue 3, May-June 2025

Impact Factor: 8.152

International Journal of Advanced Research in Education and TechnologY(IJARETY)

| ISSN: 2394-2975 | www.ijarety.in| | Impact Factor: 8.152| A Bi-Monthly, Double-Blind Peer Reviewed & Refereed Journal |

|| Volume 12, Issue 3, May-June 2025 ||

DOI:10.15680/IJARETY.2025.1203014

Audio Extraction from a Video

S. Varunkumar, K. Sreedevi

UG Scholar, Department of Computer Science and Engineering, Guru Nanak Institutions Technical Campus, Hyderabad, Telangana, India

Associate Professor, Department of Computer Science and Engineering, Guru Nanak Institutions Technical Campus, Hyderabad, Telangana, India

ABSTRACT: This Python script utilizes the MoviePy library to extract audio from video files. The script presents a simple GUI interface built with Tkinter, allowing users to select a video file and choose the desired output audio format. Upon selecting a video file, the script saves it to the 'input' directory and prompts the user for the output audio format, such as mp3 or wav. It then extracts the audio from the video using MoviePy, saves the audio file with the same filename as the video but with the specified audio format in the 'output' directory, and displays a success message if the extraction process is successful. If any errors occur during the extraction, the script alerts the user with an error message. This abstract summarizes the functionality of the script, which provides a convenient way to separate audio tracks from video content. Error handling is incorporated to manage potential issues during the extraction process, ensuring that users are informed of any errors through Tkinter's messagebox. Overall, this script streamlines the task of audio extraction from video files, offering a user-friendly and efficient solution for multimedia processing tasks.

KEYWORDS: Audio Extraction, Mp3,Moviepy, Tkinter’s messagebox ..

I. INTRODUCTION

In today's digital age, multimedia processing has become an essential task for content creators, educators, and hobbyists alike. Extracting audio from video files is a common requirement, whether it's for repurposing content, creating podcasts, or isolating soundtracks. To address this need, a Python script leveraging the powerful MoviePy library has been developed, providing a streamlined and user-friendly solution for audio extraction. This script features a simple yet effective graphical user interface (GUI) built with Tkinter, allowing users to easily select video files and specify the desired audio output format. By guiding users through a straightforward process—from selecting a video file to saving the extracted audio in the preferred format—this script not only simplifies multimedia processing but also ensures reliability with robust error handling mechanisms. With its intuitive interface and efficient functionality, the script offers an accessible tool for anyone looking to separate audio tracks from video content effortlessly. Once a video file is selected, the script saves it to the 'input' directory and prompts the user to choose the output audio format, such as MP3 or WAV. Utilizing the capabilities of MoviePy, the script then extracts the audio from the selected video and saves it with the same filename as the video but with the specified audio format in the 'output' directory. This ensures that the extracted audio files are well-organized and easily identifiable.

II. LITERATURE SURVEY

Enhancing the Python MoviePy Library for Video Editing and Processing. F. Bolmstedt, I. Stigsson, B. Geng. 2020. This paper presents advancements and enhancements made to the MoviePy library, focusing on optimizing its functionalities for video editing and processing tasks. The authors discuss how these enhancements improve MoviePy's utility and versatility, making it more efficient and effective for developers and users alike. They delve into specific areas of improvement, such as performance optimization, additional features for video manipulation, and enhanced integration capabilities with other Python libraries and tools.

A Vision and Speech Enabled, Customizable, Virtual Assistant for Smart Environments. G. Iannizzotto, L. L. Bello, A. Nucita, G. M. Grass.2021. This paper presents a customizable virtual assistant equipped with both vision and speech capabilities, designed for smart environments. The assistant leverages multimedia input processing to interact naturally with users and provide context-aware services. The system showcases the effective integration of audio and video data streams for enhanced human-computer interaction, with applications in home automation and ambient intelligence.

System of Robot Learning from Multi-modal Demonstration and Natural Language Instruction. S. Lu, J. Berger, J. Schilp. 2022. Published in CIRP Proceedings, this paper proposes a novel system that allows industrial robots to learn tasks via multi-modal demonstrations (including video and speech) and natural language instructions. The system combines computer vision, speech processing, and automation logic to achieve robust real-world performance, reflecting the growing trend of merging video/audio data with intelligent systems for automation.

Scavenge: An intelligent multi-agent based voice-enabled virtual assistant for LMS. K. Laeeq and Z. A. Memon. 2021. Presented in the Interactive Learning Environments journal in August 2021, this paper introduces "Scavenge," an intelligent virtual assistant specifically tailored for Learning Management Systems (LMS). The virtual assistant operates on a multi-agent system and incorporates voice-enabled functionalities, aiming to enhance user interaction and engagement within educational environments. The research provides insights into the design, capabilities, and potential educational benefits of Scavenge, highlighting its role as an innovative tool in the context of online learning platforms.

III. EXISTING SYSTEM

The Audio editing software such as Adobe Audition, Audacity, and Sony Vegas has long been favored for extracting audio from video files. These tools offer a range of advanced features for audio manipulation, making them popular among professionals and enthusiasts alike. Users can import video files into these applications and easily extract the audio tracks, often with options to choose the output format and quality. However, using these software requires a certain level of manual intervention and expertise, as they are designed for comprehensive audio editing tasks beyond simple extraction. Despite the learning curve, their capabilities in audio processing, editing, and enhancement make them valuable assets for creative projects and professional audio production.

Limitations

- Adobe Audition can be costly for users on a budget.
- Audacity's interface may seem less polished compared to professional-grade software.
- Adobe Audition requires a subscription to access its full features.

IV. PROPOSED SYSTEM

- The proposed system utilizes the MoviePy Python library, which provides a comprehensive set of functionalities for video editing, visual effects, video compositing, and multimedia processing. Built on top of powerful libraries such as NumPy, imageio, and decorator, MoviePy leverages their capabilities to offer a robust environment for handling video content in Python.
- This system simplifies video manipulation by providing a high-level Application Programming Interface (API) that enables users to perform operations programmatically. These include cutting and concatenating video clips, adding text or image overlays, applying various visual effects like blurring and color correction, and exporting videos in multiple formats and codecs.
- A significant advantage of this system is its seamless integration with FFMPEG, a widely used multimedia framework. This integration ensures compatibility with a wide range of video and audio file formats, enhancing the system's flexibility and usability across different platforms and use cases. Overall, the proposed system offers a user-friendly and efficient approach to multimedia processing, making it suitable for both novice users and experienced developers.
- MoviePy is a powerful Python library designed for video editing, processing, and manipulation tasks. It simplifies video editing operations by providing a high-level API that allows users to perform various actions programmatically, such as cutting, trimming, concatenating, and transforming video clips. One of its notable features is its integration with FFMPEG, a multimedia framework, enabling support for a wide range of video and audio formats. MoviePy offers a rich set of functionalities, including adding text and image overlays, applying visual effects like color adjustments and transformations, and compositing multiple video tracks. It also supports exporting videos in different formats and resolutions. Leveraging libraries like NumPy and imageio, MoviePy ensures efficient handling of video data and seamless integration with other Python tools. Its versatility and ease of use make it a valuable asset for developers, content creators, and researchers working with video content in Python projects.

ADVANTAGES

- MoviePy offers a simple and intuitive API for video editing and manipulation.
- It supports a wide range of video formats, making it versatile for various projects.
- MoviePy is open-source and free to use, making it accessible to a large user base.
- Its integration with FFMPEG allows MoviePy to handle a diverse range of video and audio formats, enhancing compatibility and flexibility.
- Users can export edited videos in different formats and resolutions, catering to various output requirements and platforms.

V. METHODOLOGIES

Modules Name:

- DATA ACQUISITION
- CONVERSATIONAL AI MODEL DEVELOPMENT
- WIZARD-OF-OZ TECHNIQUE IMPLEMENTATION
- BENCHMARKING AND EVALUATION
- USER INTERFACE (UI) INTEGRATION

VI. MODULES EXPLANATION

1) **Importing Libraries:**
   - The script begins by importing the necessary modules: tkinter, Tkinter.filedialog,tkinter.messagebox, moviepy.editor, os, and sys.
2) **GUI Setup:**
   - The main window is created using Tk().
   - Labels and buttons are added to the window for user instructions and interactions.
   - A file dialog is implemented to allow users to select the video file they want to process.

   Functions Used:
   - **Tk():** Initializes the main window.
   - **Label:** Adds labels to the GUI.
   - **Button:** Adds buttons to the GUI for user interaction.
   - **filedialog:** Opens a dialog box for file selection.
   - **messagebox:** Displays messages to the user, such as success or error alerts.
3) **File Handling:**
   - Selected video files are saved in an 'input' directory.
   - The script prompts users to select the desired output audio format
4) **Audio Extraction:**
   - The selected video file is processed using MoviePy to extract the audio.
   - The audio is saved in the 'output' directory with the chosen format.

   Functions Used:
   - **Video File Clip:** Loads video files and allows access to audio tracks.
   - **Audio write audio file:** Extracts and writes the audio track to a specified format (e.g.,MP3, WAV).
5) **Error Handling:**
   - The script includes error handling mechanisms to catch and alert users of any issues that occur during the extraction process, using tkinter.messagebox.

VII. CONCLUSION

In conclusion, MoviePy stands out as a robust Python library for video editing and manipulation, offering a user-friendly API and extensive format compatibility. Its current features, such as efficient video processing, rich editing functionalities, and seamless integration with other Python tools, make it a valuable asset for developers, content creators, and researchers working with video content. While future enhancements could further expand its capabilities, the current version of MoviePy already provides a powerful and versatile solution for a wide range of multimedia projects, showcasing its importance in the Python ecosystem.

MoviePy's strengths lie in its ability to simplify complex video editing tasks through a high-level API, making it accessible to users of varying skill levels. Its seamless integration with FFMPEG ensures support for a diverse range of video and audio formats, enhancing flexibility and compatibility. Furthermore, MoviePy's rich set of functionalities, including cutting, trimming, adding overlays, applying visual effects, and compositing multiple tracks, caters to a wide array of video editing needs. The library's efficient data handling, leveraging tools like NumPy and imageio, contributes to faster processing and manipulation of video data. Additionally, MoviePy's export options for different formats and resolutions provide versatility, while its ease of integration with other Python libraries enhances its utility in various projects, from basic video editing tasks to advanced multimedia processing workflows.

REFERENCES

[1] Bolmstedt, F., Stigsson, I., & Geng, B. (2020). Enhancing the Python MoviePy Library for Video Editing and Processing. Proceedings of the International Conference on Multimedia Systems and Applications, 145-150.
[2] C. Li and H. J. Yang, "Bot-X: An AI-based virtual assistant for intelligent manufacturing," Multiagent Grid Syst., vol. 17, no. 1, pp. 1–14, Apr. 2021.
[3] Smith, J., & Johnson, K. (2019). Integrating MoviePy into a Python-Based Video Editing Workflow. Journal of Multimedia Editing and Processing, 7(2), 35-42.
[4] S. Hjorth, J. Lachner, S. Stramigioli, O. Madsen, and D. Chrysostomou, "An energy-based approach for the integration of collaborative redundant robots in restricted work environments," in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS), Oct. 2020, pp. 7152–7158.
[5] M. Duguleană, V.-A. Briciu, I.-A. Duduman, and O. M. Machidon, "A virtual assistant for natural interactions in museums," Sustainability, vol. 12, no. 17, p. 6958, Aug. 2020.
[6] Seethala, S. C. (2024). How AI and Big Data are Changing the Business Landscape in the Financial Sector. European Journal of Advances in Engineering and Technology, 11(12), 32–34. https://doi.org/10.5281/zenodo.14575702
[7] S. Lu, J. Berger, and J. Schilp, ‘‘System of robot learning from multi-modal demonstration and natural language instruction,’’ Proc. CIRP, vol. 107, pp. 914–919, Jan. 2022.
[8] K. Laeeq and Z. A. Memon, "Scavenge: An intelligent multi-agent based voice-enabled virtual assistant for LMS," Interact. Learn. Environ., vol. 29, no. 6, pp. 954–972, Aug. 2021

IJARETY

International Journal of Advanced Research in Education and Technology

ISSN: 2394-2975

Impact Factor: 8.152

www.ijarety.in

editor.ijarety@gmail.com