Language Translation with RaspberryPI

Barath kumar J
CSE 2021 - 2025
Amrita Vishwa Vidyapeetham
Coimbatore, India
barathkumarj001@gmail.com

Rohith kumar Chunduru
CSE 2021 - 2025
Amrita Vishwa Vidyapeetham
Coimbatore, India
rohitkumarchunduru@gmail.com

Harish S
CSE 2021 - 2025
Amrita Vishwa Vidyapeetham
Coimbatore, India
harishsridharan03@gmail.com

Shri Praveen S M
CSE 2021 - 2025
Amrita Vishwa Vidyapeetham
Coimbatore, India
shripraveen47@gmail.com

Sudharsan B
CSE 2021 – 2025
Amrita Vishwa Vidyapeetham
Coimbatore, India
bsudharsan27@gmail.com

***Abstract*— This paper introduces a Raspberry Pi-based language translation system using the Google Translate API. The system captures user speech, translates it into a chosen language, and converts the translated text into speech. The implementation showcases the practicality of real-time multilingual communication on the Raspberry Pi, emphasizing its versatility and efficiency for compact language processing applications.**

I. Introduction

In recent years, the rise of compact and affordable computing devices has opened the door to innovative solutions across various fields. Notably, the Raspberry Pi, a credit card-sized single-board computer, has gained attention for its adaptability and versatility. This document explores the creation of a language translation system on the Raspberry Pi, specifically focusing on real-time multilingual speech translation using the Google Translate API.

The challenges posed by language barriers in today's globalized society can impede effective communication and collaboration. This study tackles these challenges by leveraging the capabilities of the Raspberry Pi to develop a portable and cost-effective solution for on-the-go language translation. Through the integration of the Google Translate API, the system allows users to interact seamlessly with the device through speech recognition and receive translated output via synthesized speech.

The Raspberry Pi's compact design, combined with cloud-based translation services, presents a practical approach to overcoming resource constraints while maintaining computational efficiency. This document outlines the design, implementation, and evaluation of the proposed language translation system, highlighting the adaptability and accessibility of the Raspberry Pi in facilitating real-time multilingual communication.

Through this exploration, our goal is to contribute to the expanding body of research on edge computing applications, showcasing the potential of compact devices like the Raspberry Pi to address language barriers and enhance communication in diverse contexts. The subsequent sections delve into the methodology, implementation details, and performance evaluations, providing insights into the feasibility and practicality of the proposed solution.

II. Components Used

A. *Raspberry Pi 3:*

The Raspberry Pi 3 is a single-board computer developed by the Raspberry Pi Foundation. Released in February 2016, it is part of the Raspberry Pi series and represents a significant advancement in terms of performance and features. The key attributes of the Raspberry Pi 3 include a 1.2 GHz quad-core ARM Cortex-A53 processor, 1 GB of RAM, built-in Wi-Fi (802.11n), and Bluetooth 4.1 capabilities. With its diverse range of applications, from DIY projects to educational endeavors, the Raspberry Pi 3 has become a go-to choice for enthusiasts and developers alike.

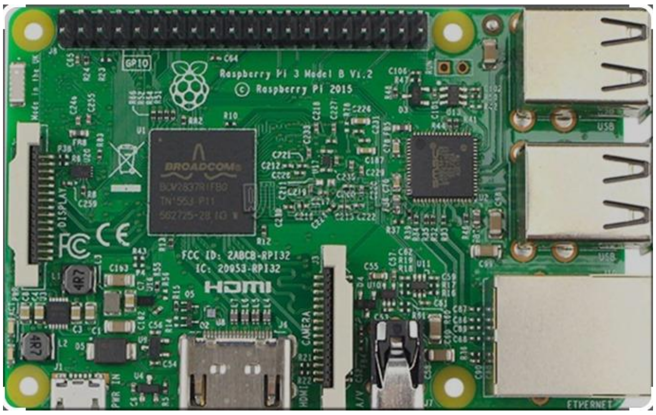

Fig. 1. Raspberry pi 3

B. *ST7735 SPI Display for Integration:*

The ST7735 is a popular TFT (Thin-Film Transistor) display controller, and when coupled with an SPI (Serial Peripheral Interface) communication protocol, it becomes a suitable choice for interfacing with single-board computers like the Raspberry Pi. These displays are characterized by their small physical footprint, making them practical for portable projects and applications with space constraints. The integration of ST7735 TFT SPI displays in language translation systems serves multiple purposes. Firstly, these displays act as a visual interface, providing users with real-time feedback on the translation process. Whether displaying text translations or graphical elements, the ST7735 display contributes to creating a more interactive and user-friendly experience.

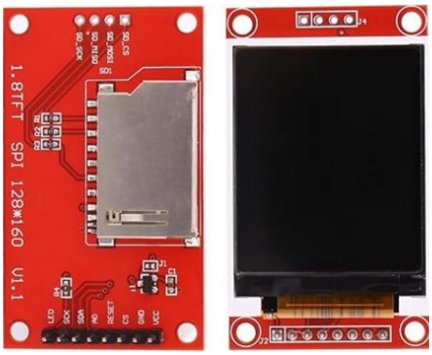

Fig. 2. ST7735 TFI SPI

C. USB Microphone for Speech Input:

The incorporation of a USB microphone into a language translation system significantly enhances its capabilities, particularly in the domain of speech recognition. By providing a dedicated audio input channel, USB microphones enable the system to capture spoken language with clarity, facilitating accurate and reliable recognition. USB microphones play a crucial role in the speech recognition component of the language translation system. These microphones are designed to pick up audio signals with minimal interference, ensuring that spoken words are accurately captured. The data collected by the USB microphone is then processed using speech recognition algorithms to convert the spoken language into machine-readable text.

Fig. 3. USB Microphone

D. Boat Stone 193 for synthesized Speech Output:

The Boat Stone 193 is a Bluetooth speaker known for its compact design, portability, and wireless capabilities. It is equipped with Bluetooth technology, allowing seamless audio streaming from compatible devices. The integration of such Bluetooth speakers, including the Boat Stone 193, into language translation systems on the Raspberry Pi introduces a new dimension by enabling synthesized speech output.

The integration of Bluetooth speakers facilitates synthesized speech output in language translation systems. After a text translation is generated, the Raspberry Pi can use Text-to-Speech technology to convert the translated text into audible speech. The Boat Stone 193, as a Bluetooth-enabled device, serves as the medium fordelivering this synthesized speech, allowing users to hear the translated content in real-time.

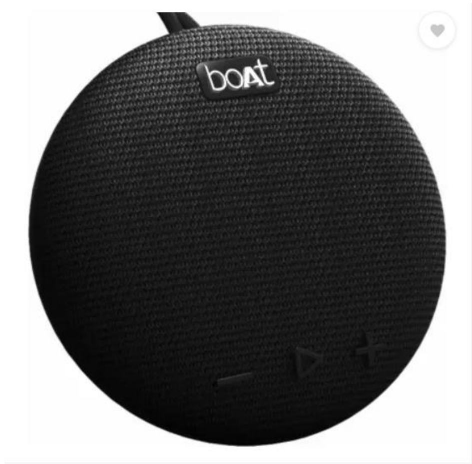

Fig. 4. Boat Stone 193

E. Jumper Wires for Peripheral Connections:

Jumper wires play a crucial role in establishing reliable connections between the Raspberry Pi and various peripherals, including components such as the ST7735 TFT SPI display, USB microphone, and other hardware relevant to language translation applications. These wires are flexible connectors with male or female ends, enabling seamless linking of different elements within a project. Jumper wires are highly versatile, facilitating the creation of custom wiring configurations. Their flexibility allows users to adapt and customize the connections based on the specific requirements of the language translation system. This versatility is particularly advantageous in projects where space constraints or unique

Fig. 5. Jumper Wires

III. START UP PROTOCOL FOR RASPBERRY PI OS

A. *About the OS:*

The Raspbian 64-bit Legacy OS orchestrates an optimization strategy to exploit the heightened computational capabilities inherent in 64-bit systems. This deliberate adaptation translates into marked improvements in performance, particularly evident in memory-intensive applications. Furthermore, the OS ensures heightened compatibility with an expansive array of software engineered to capitalize on the advantages offered by 64-bit architecture.

B. *Empowering Users with 64-bit Computing:*

A testimony to the remarkable adaptability of the Raspberry Pi platform, the Raspbian 64-bit Legacy OS empowers users to harness the full spectrum of benefits offered by 64-bit computing. Through this OS, users experience a transformative leap in processing capabilities while simultaneously enjoying a level of compatibility that ensures a smooth transition within the well-established Raspbian ecosystem. This unique blend of innovation and continuity exemplifies the evolution and versatility inherent in the Raspbian 64-bit Legacy operating system.

C. *Booting process:*

The Raspberry Pi 64-bit Legacy OS can be booted from a variety of sources, including an SD card, a USB drive, or a network location.

The following steps outline the general booting process:

1. Insert the SD card or USB drive containing the Raspbian 64-bit Legacy OS into the Raspberry Pi.
2. Connect the Raspberry Pi to a power source.
3. The Raspberry Pi will automatically boot from the SD card or USB drive.
4. Once the boot process is complete, the Raspberry Pi will display the Raspbian 64-bit Legacy OS login screen.

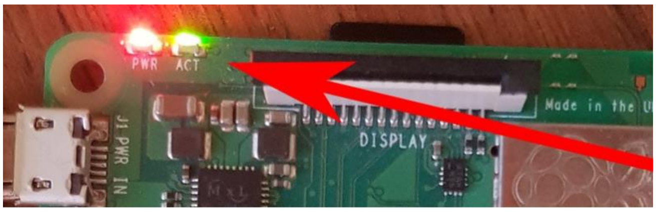

Fig. 6. Raspberry Pi 3 with red and green LEDs indicating boot status.

IV. CONNECTION SETUP

A. *Connection of Pi and ST7735 TFI SPI Display:*

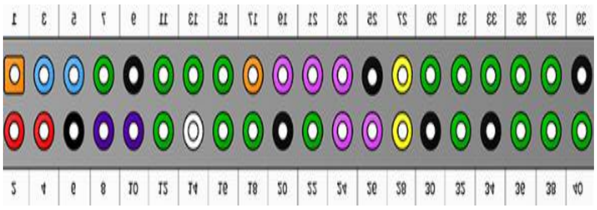

Fig. 7. Pin diagram of Raspberry pi 3

Fig. 8. Pin diagram of ST7735 TFI SPI Display

TABLE I

Raspberry Pi	ST7735 TFI SPI	Purpose	Connection
3.3V Power (1st Pin):	LED (Light Emitting Diode)	Provides power to the LED backlight of the display.	Linked to the 3.3V power source on the Raspberry Pi (1st pin).
SPI Clock(Serial Peripheral Interface) (23rd Pin)	SCK (Serial Clock)	Transmit clock signals for Aligned data transfer.	Wired to GPIO (General Purpose Input/Output) on the Raspberry Pi (23rd pin)
SPI MOSI(Master Out Slave In) (19th Pin)	SDA (Serial Data Line)	Transmits data from the Raspberry Pi to the display	Connected to GPIO IO (MOSI) on the Raspberry Pi (19th pin)
GP1018 (12th Pin)	AO(Analog 0)	Selects between command and data for the display.	Linked to GPIO 18 on the Raspberry Pi (12th pin).

Raspberry Pi	ST7735 TFI SPI	Purpose	Connection
GP1023 (16th Pin)	RESET	Resets the display to its default state.	Wired to GPIO 23 on the RaspberryPi (16th pin).
GP1024 (18th Pin)	CS(Chip Select)	Enables or disables the display for data communication.	Connected to GPIO 24 on the RaspberryPi (18th pin).
Ground (39th Pin)	GND (Ground)	Serves as the common ground reference.	Connected to the ground (GND) pin on the RaspberryPi (39th pin).
5V Power (2nd Pin):	VCC (Voltage at the Common Collector)	Provides power to the display.	Wired to the 5V power source on the RaspberryPi (2nd pin).

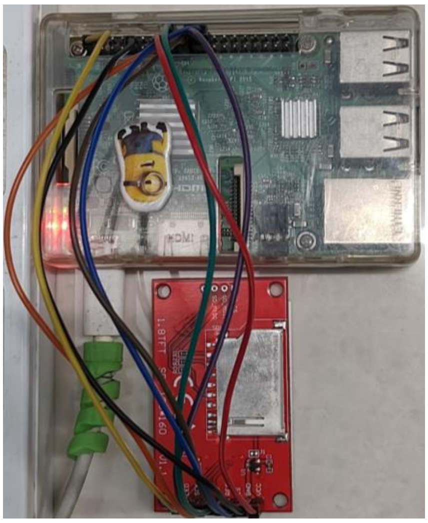

Fig. 9. Connection of Pi and ST7735 TFI SPI Display

B. *Connection of Pi and USB Microphone:*

The USB microphone is connected to the Raspberry Pi via a standard USB port. The USB port on the Raspberry Pi serves as the interface for the USB microphone, allowing for a simple and convenient connection. This connection is established by inserting the USB plug of the microphone into an available USB port on the Raspberry Pi. The USB interface facilitates both power and data communication between the Raspberry Pi and the USB microphone, enabling seamless audio input functionality. This setup adheres to industry-standard USB protocols, ensuring compatibility and ease of integration within the Raspberry Pi ecosystem. The USB connection is a key component in enabling the microphone to capture and transmit audio data to the Raspberry Pi for further processing and analysis.

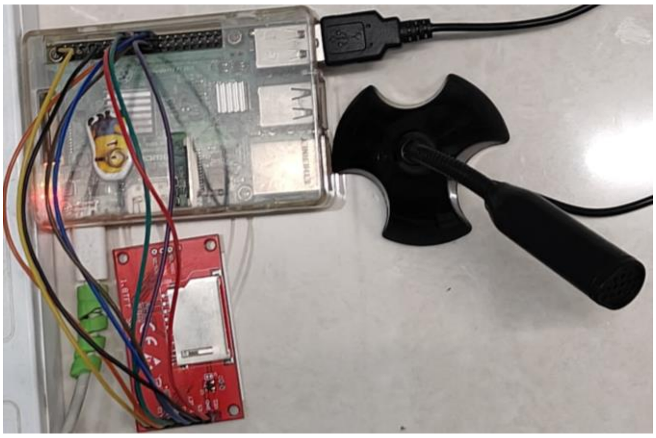

Fig. 10. Connection of Pi and USB Microphone

C. *Connection of Pi and Boat Stone 193:*

The audio output port of the Raspberry Pi serves as the point of connection for the Boat Stone 193 speaker. The connection is established by utilizing a standard 3.5mm audio cable that links the audio output jack of the Raspberry Pi to the corresponding input port on the Boat Stone 193. This audio connection adheres to industry standard practices, ensuring compatibility between the Raspberry Pi and the external speaker. The 3.5mm audio cable functions as the conduit for transmitting audio signals from the Raspberry Pi to the Boat Stone 193, facilitating the reproduction of sound output.

The connection is straightforward and requires inserting the 3.5mm plug into the audio output port on the Raspberry Pi, and the other end into the input port on the Boat Stone 193. This configuration allows the Raspberry Pi to deliver audio output signals to the Boat Stone 193 speaker, enabling the playback of audio content, system sounds, or other auditory feedback.

The use of a widely accepted audio interface ensures a reliable and standardized connection, contributing to the seamless integration of the Raspberry Pi with the external audio device.

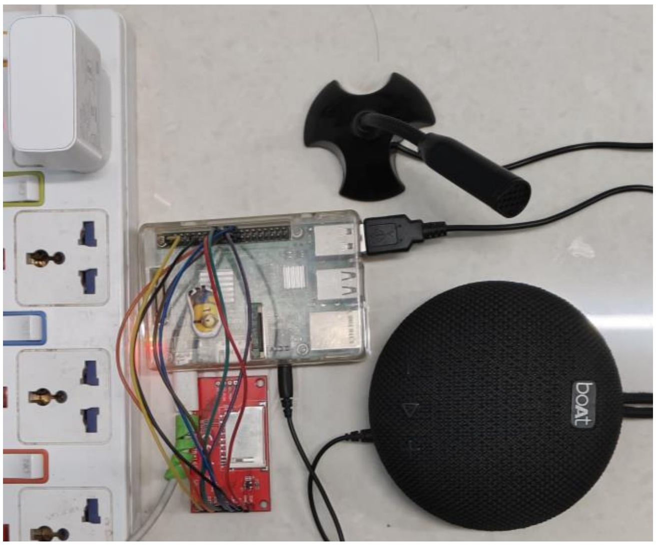

Fig. 11. Connection of Pi and Boat Stone 193

V. Methodlogy

A. *User Input:*

1. Users articulate sentences into the USB microphone
2. Translation processes are initiated upon detecting user input.

B. *Language Detection:*

1. Automated language detection identifies the input language.
2. The detected language is presented on the console

C. *Target Language Input:*

1. Users specify the target language through the microphone.
2. System processing prepares for translation

D. *Translation Process:*

1. The Google API , a third-party service , is employed to perform the translation based on the provided inputs.
2. The raspberry Pi 3 microcontroller manages the communication with the Google API , sending the sentence for translation and receving the translated output.

E. *Console Display:*

The translated text is first displayed on the console allowing for easy verification and reference.

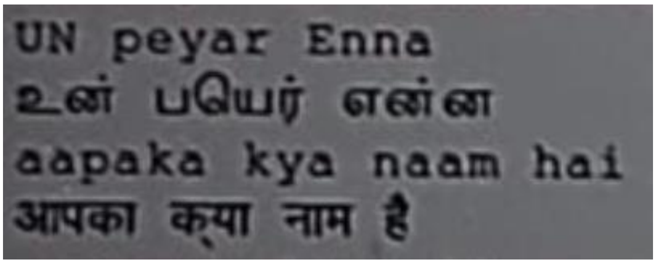

Fig. 12. Console output

Four lines of output provides comprehensive information.

1. Thanglish representation of the input sentence.
2. Font is of the actual language of the input sentence.
3. Thanglish representation of the output sentence.
4. Font is of the actual language that is to be translated.

This Thanglish representation is done because if someone does not know his language or the language in which he is going to translate to read or write, he can make use of the thanglish.

F. *ST7735 SPI Display:*

Simultaneously, the translated output is sent to the ST7735 TFI SPI Display, providing a visual representation of the translated text.

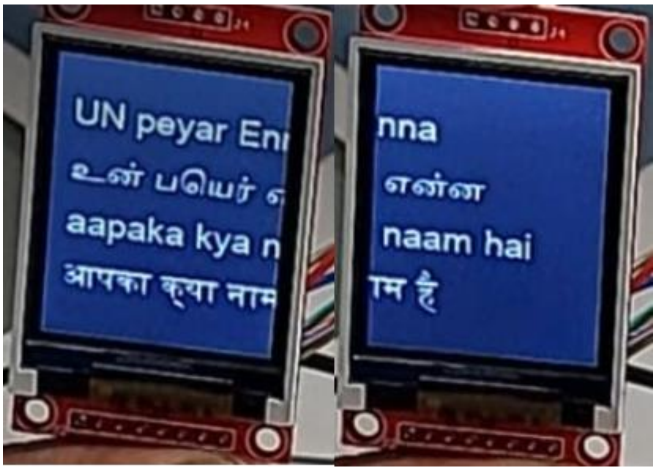

Fig. 13. ST7735 SPI Display output

Suppose if the user is deaf he can still make use of this display to view input sentence and the translated sentence. Even though he cannot hear the audio output he can still see what is happening on the screen. This can be helpful for understanding visual information.

G. Boat Stone 193 Output:

The Boat Stone 193 speaker then audibly communicates the translated sentence, ensuring that users receive both visual and auditory feedback.

H. User Interaction:

Following the translation process, the system engages users:

1. "Do you want to hear again?" (input: "Yes" or "No")
2. If "Yes," the translated text is reiterated through the speaker.
3. If "No", the system gracefully responds with "Thank you, have a nice day," and the system concludes.

For detailed video explanation follow this youtube link.

VI. CONCLUSION

In summary, the implementation of our integrated multi-modal language translation system represents a significant stride towards creating a user-centric and versatile language translation solution. By seamlessly combining hardware components and harnessing the capabilities of the Google API, our system addresses the diverse preferences of users, providing a comprehensive experience that extends beyond the traditional boundaries of translation services.

The emphasis on both visual and auditory modalities enhances the accessibility and usability of the system. The inclusion of a USB microphone, ST7735 SPI display, and Boat Stone 193 speaker reflects our commitment to accommodating various user preferences. This holistic approach not only caters to individuals with different learning and communication styles but also ensures that the system can be utilized in a range of environments and scenarios.

Moreover, the synergy achieved between the Raspberry Pi 3 microcontroller and the Google API serves as the backbone of the system's robustness. The efficiency in managing communication and processing translations not only guarantees accurate and timely results but also positions the system for scalability and adaptability. This foundation, rooted in advanced technology, reinforces the inclusivity of the solution, making it a versatile tool for users across various linguistic landscapes.

VII. FUTURE DEVELOPMENTS

The future development of our system could involve refining languagedetection algorithms to improve accuracy. A more sophisticated language detection mechanism would not only enhance the precision of translations but also contribute to a smoother and more intuitive user experience.

Integration of supplementary translation services beyond the Google API is another potential direction for future development. This would offer users a choice of translation engines, allowing for flexibility and adaptation to evolving technological landscapes.

User feedback will play a pivotal role in shaping the trajectory of this project. Continuous refinement based on real-world user experiences will ensure that the system evolves to meet the dynamic needs and expectations of its users. Usability studies and user-centric design principles will guide future enhancements, fostering a system that remains relevant and effective in various contexts.

The system's success in its current scope is a testament to its potential for growth and adaptation. By staying attuned to user feedback and embracing emerging technologies, the multi-modal language translation system is poised to make meaningful contributions to the evolving landscape of language translation solutions.

VIII. REFERENCES

[1] ST7735 https://github.com/degzero/Python_ST7735/tree/master

[2] Google API Translation https://www.geeksforgeeks.org/language-translator-using-google-api-in-python/

[3] gTTS https://www.geeksforgeeks.org/convert-text-speech-python/

[4] https://www.displayfuture.com/Display/datasheet/controller/ST7735.pdf

[5] https://www.youtube.com/watch?v=SYdGNpfLxKw

[6] https://github.com/degzero/Python_ST7735/tree/master