

Communicative Voice Assistant

Dr. S. V. Viraktamath¹, Anusha G Sheelvant², B N Saritha Reddy³, Sumayya Darugar⁴

¹(Department of Electronics and Communication Engineering, SDM College of Engineering and Technology, Dharwad
Email: svvmath@gmail.com)

²(Department of Electronics and Communication Engineering, SDM College of Engineering and Technology, Dharwad
Email : anushasheelvant2003@gmail.com)

³(Department of Electronics and Communication Engineering, SDM College of Engineering and Technology, Dharwad
Email : sarithareddy2829@gmail.com)

⁴(Department of Electronics and Communication Engineering, SDM College of Engineering and Technology, Dharwad
Email : sumayyadarugar29@gmail.com)

Abstract:

Communicative voice assistants have become a key part of our daily lives, due to improvements in AI and machine learning. These assistants, which used to be simple text-to-speech tools, can now understand and respond to voice commands more effectively. They are found in devices like smart speakers and smartphones, and are used in areas like healthcare, home automation, education, and cars to make tasks easier and more accessible. While there are still challenges, such as privacy issues, voice assistants have the potential to make everyday life smarter and more inclusive. This paper explores the rise of AI-powered voice assistants, covering their types, and also presents an implemented and tested system that integrates facial recognition and voice interaction to enhance user accessibility and security.

Keywords — Natural Language Processing (NLP), Optical Character Recognition (OCR), Speech Recognition, OpenCV.

I. INTRODUCTION

The rapid advancement of AI and machine learning has propelled speech recognition technology into widespread use, making it an integral part of daily life. Since speaking is faster and more natural than typing, voice technology has found applications across industries. This evolution is largely driven by machine learning and neural networks, shifting human-computer interaction towards automation. Voice assistants from Google, Apple and Microsoft empower users, including the elderly and disabled, to perform tasks such as device control, navigation and even vehicle management for visually impaired individual.

Initially limited to basic text-to-speech systems, modern voice assistants are now robust conversational AI platforms. By incorporating

technologies like machine learning and neural networks, they have evolved to understand context and intent, delivering adaptive and meaningful

interactions. These systems are integral to daily tasks such as managing emails and calendars, showcasing their expanding role in modern life [1].

A comparative analysis indicates regional differences in how users engage with voice assistants. For example, Chinese users prioritize lifestyle features, while others focus on work-related tasks. Nevertheless, these technologies consistently improve quality of life worldwide. Voice portals further illustrate this trend, enabling users to access web-based information through voice commands. This innovation has created new business opportunities and improved accessibility to digital services.

Voice portals allow users to access information online using voice commands anytime and anywhere, due to advanced speech recognition. This has created new opportunities for businesses by making their services more accessible, especially with the growing use of smartphones and smart home devices. While voice portals are becoming a popular way to access information, there are still some challenges and limitations that need to be worked out [2].

II. TYPES OF VIRTUAL ASSISTANCE

Voice assistants can be categorized into two main types: device-based and software-based. Device-based voice assistants include smart speakers like Amazon Alexa and Google Home, mobile-based voice assistants such as Siri and Google Assistant, and wearables like smartwatches and smart earbuds. On the other hand, software-based voice assistants refer to Voice AI integrated into applications, such as chatbots and virtual agents, which provide voice-enabled interaction within software environments.

A. Device-based voice assistant

1) **Amazon Alexa:** Amazon Alexa, introduced with the Echo smart speaker, is now available on various devices, offering over 100,000 voice-controlled skills. Users activate it by saying "Alexa," and the system uses Automatic Speech Recognition (ASR) and Natural Language Understanding (NLU) to process requests and respond accordingly.

2) **Google Assistant:** Google Assistant, developed by Google, is built into devices like smartphones and smart TVs, providing voice control and multi-language support. Activated with "Hey, Google" or "OK, Google," it processes audio using Speech-to-Text, Natural Language Processing, and Understanding to interpret commands. The request is handled by the Conversation API, which responds via Text-to-Speech. Developers can also add custom features using the Google Actions SDK [3].

3) **Amazon Alexa:** Amazon Alexa, introduced with the Echo smart speaker, is now available on various devices, offering over 100,000 voice-controlled skills. Users activate it by saying "Alexa," and the system uses Automatic Speech Recognition (ASR) and Natural Language Understanding (NLU) to process requests and respond accordingly.

Figure 1 shows the timeline of the various Device based Voice Assistants.



Fig. 1 Timeline of Device based Voice Assistants [4].

B. Software-based voice assistants

4) **Chatbots:** Virtual assistants like Siri and Alexa, along with chatbots, are changing how we interact with technology. They offer convenient, hands-free access to information and instant responses in various industries. While challenges remain in understanding complex queries and ensuring privacy, ongoing advancements will enhance their personalization and integration with smart devices, providing efficient and human-like interactions [5].

5) **Virtual agents:** An Intelligent Virtual Assistant (IVA) or Intelligent Personal Assistant (IPA) is a software agent that performs tasks based on voice commands or questions. Often referred to as a chatbot, it can interpret human speech and respond using synthesized voices. Virtual assistants can manage tasks like answering queries, controlling smart devices, playing media, and handling email, to-do lists, and calendars. Some chat programs are for entertainment, but more advanced IVAs focus on productivity and task automation [6].

III. SYSTEM ARCHITECTURE

The architecture of a virtual assistant is shown in figure 2.

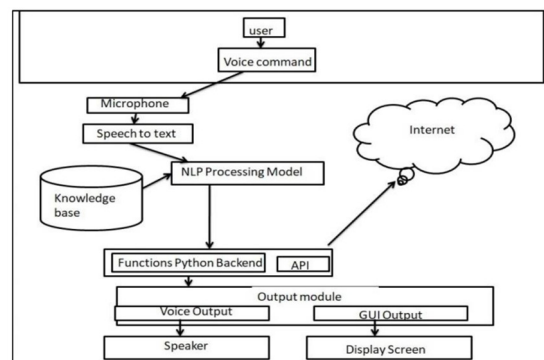


Fig. 2 Timeline System Architecture [7].

It shows the flow from the User Interface Layer, where input is captured, through modules like Speech Recognition and NLU for understanding, to the Dialog Management System for maintaining context. The Task Execution Engine handles backend operations, while the Knowledge Base and Memory provide personalized and informed outputs. Finally, Multimodal Output and Integration with External Services ensure comprehensive and versatile user responses [7].

A. Detailed workflow

The detailed workflow of voice assistant can be described into a series of steps as shown in figure 3.

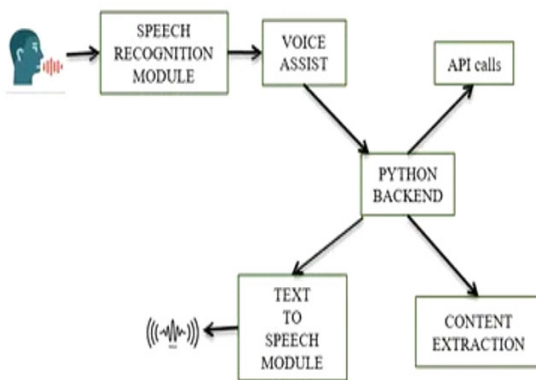


Fig. 3 Detailed Workflow of voice assistant [8].

- **Speech Recognition:** The voice data provided to the system is through Google online speech recognition which uses voice and converts it into text. First, the voice data is kept for some while and then it is transmitted to Google’s cloud for further analysis. The converted text is forwarded to the central processor.
- **Python Backend:** After the speech has been ploughed into appropriate text form, how the text was also examined as to whether the input was of an application program interface action, context extraction or a system one. Feedback is then given to the user.
- **API Calls:** Application programming interface, abbreviated API, provides interlinking functionalities between two apps. Simply put, an application sends a request to another and

gets a response, hence communication between various software parts is possible.

- **Context Extraction (CE):** Context extraction entails the automatic collection of structured information from texts that can be unstructured or semi-structured. Such tasks are frequently accomplished through the use of natural language processing (NLP). Monetization of CE can also include visuals such as information obtained from images or sound or even videos.
- **System Calls:** A system call can be described as a service request a process makes to the operating system, wherein a process asks the operating system for permissions to use resources, spawn new processes, or run tasks. It serves as the interface in a program and a kernel of an Operating System [8].

B. Sequence diagram

The Sequence Diagram for the Voice communicative Assistant is shown in figure 4. The user begins by providing a voice command, which is captured by the microphone and sent to the interpreter for processing. Next, the web scraper retrieves relevant answers and solutions from the web. The obtained solution is then passed to the speaker, which uses a text-to-speech library to convert the text into audio and delivers the results in voice format [9].

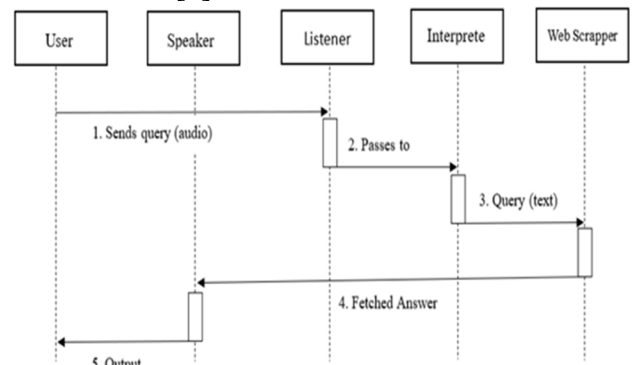


Fig. 4 Sequence Diagram [9].

IV. TYPES OF VIRTUAL ASSISTANCE

The working of Virtual Assistant uses following principles:

A. Natural Language Processing:

Natural Language Processing (NLP) is essential for virtual assistants to understand and respond to human language. It explains how NLP processes speech or text input, improving the accuracy and context of responses. For virtual assistants, this means better understanding user commands, handling complex phrases, and providing accurate answers. Advances in NLP, such as deep learning, make virtual assistants smarter and more reliable. Additionally, the combination of human input and machine learning enhances the assistant’s ability to improve over time [10]. Figure 5 shows the five steps involved in Natural Language Processing.

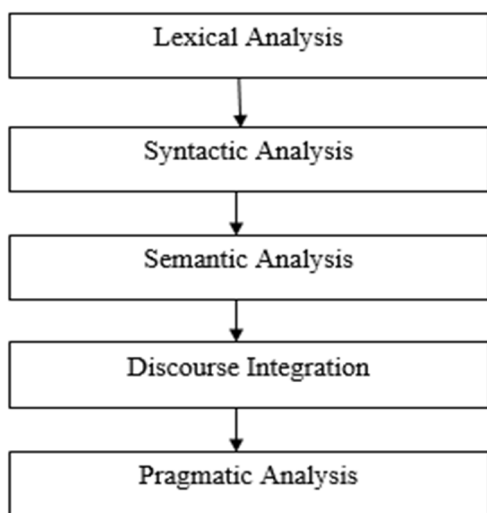


Fig. 5 Steps involved in Natural Language Processing [10].

B. Automatic Speech Recognition:

To understand command according to user’s input. The figure 6 shows the working process of Speech Recognition.

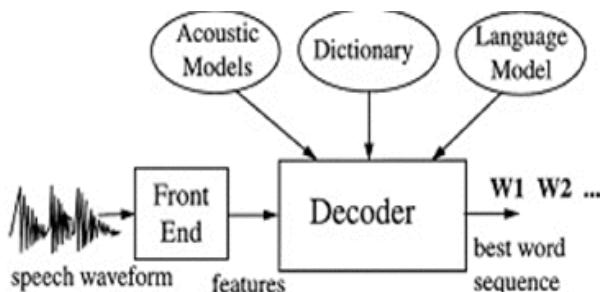


Fig. 6 Working of Speech Recognition [11].

C. Artificial Intelligence:

A voice assistant uses artificial intelligence to interact with users, understand their behaviour, and store information about preferences and relationships. It reasons, learns from experiences, solves problems, and adapts to new situations. With natural language processing, it enables seamless communication, allowing users to perform tasks and access information using voice commands. Figure 7 shows the various functions performed by AI systems.

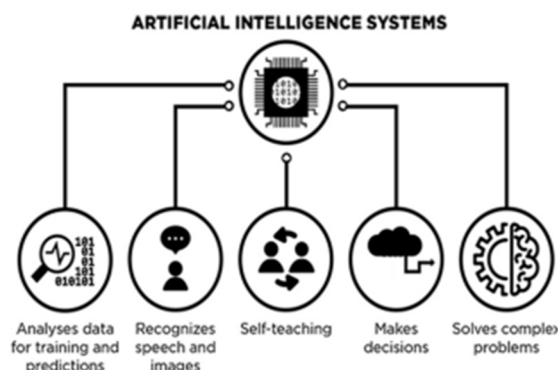


Fig. 7 Various functions of AI systems [11].

V. IMPLEMENTATION

The voice assistant system was developed using a combination of Python, JavaScript, HTML/CSS, and Tesseract.js, integrated into a desktop and web-based environment. The following technologies and libraries were used in various modules of the system:

A. Programming Languages and Frameworks

1) **Python:** Python was used for backend logic, including face authentication using OpenCV and ORB algorithms, as well as speech processing with libraries like pytsx3 for offline text-to-speech. Modules such as os, time, and web browser supported file handling, timing, and voice-command-based website access.

2) **JavaScript:** JavaScript played a crucial role in enabling browser-based features. The Speech Recognition API converted user voice commands into text, while the Speech Synthesis API enabled the system to respond with spoken feedback, creating an interactive experience.

3) **HTML/CSS:** Used to structure and style the web interface. HTML provided the layout for components such as the dashboard, help section, and product search bar, while CSS ensured a visually appealing and responsive design.

4) **Tesseract.js**: It is a JavaScript-based OCR library, was employed to extract text from uploaded images, allowing users to interact with visual content. The virtual keyboard, language toggle buttons, and other dynamic interface elements were also developed using JavaScript.

B. Libraries and Functionalities

1) **CV2 (OpenCV)**: OpenCV (cv2) was used for face detection and image processing. It captured webcam input, detected facial features using Haar cascades, and matched facial descriptors through the ORB (Oriented FAST and Rotated BRIEF) algorithm.

2) **OS**: The os module facilitated file and directory operations such as checking the existence of user image files, managing paths during image capture and storage, and ensuring seamless access to resources across the system.

3) **Time**: This module handled the timing logic within the assistant, such as implementing delays for smoother speech delivery and setting timeouts during authentication processes to prevent long wait times.

4) **pyttsx3**: pyttsx3 is a text-to-speech conversion library that operates offline. It was used to convert system-generated text into spoken words, enabling the assistant to provide audible feedback to the user, even without an internet connection.

C. JavaScript APIs and libraries

1) **Speech Recognition API**: This browser-native API enabled voice command functionality by converting spoken input from the user into text. It allowed the assistant to interpret and act on commands without requiring manual input, supporting hands-free interaction and improving accessibility, especially for visually impaired users.

2) **The Speech Synthesis API**: Used for real-time voice feedback, this API converted the assistant’s text responses into speech. It enhanced the user interface by reading out responses, confirmations, and search results directly through the browser’s speaker.

3) **Tesseract.js**: Tesseract.js is a JavaScript library for Optical Character Recognition (OCR). It was used to scan and extract text content from uploaded image files (e.g., documents or screenshots) and display it to the user.

VI. RESULTS

The voice assistant system was developed using a combination of Python and web technologies to ensure platform independence, user accessibility, and a secure, interactive interface. The implementation consists of several integrated modules.

A. Face Authentication:

To ensure secure access, a multi-view face authentication system was implemented using OpenCV. During registration, three images (front, left, and right profiles) of the user are stored. The ORB algorithm extracts key points from stored and live camera feeds, which are then compared using a Brute-Force Matcher. If the matched key points exceed a set threshold, access is granted. This process ensures only authorized users can use the assistant. The output is shown in Figure 8.

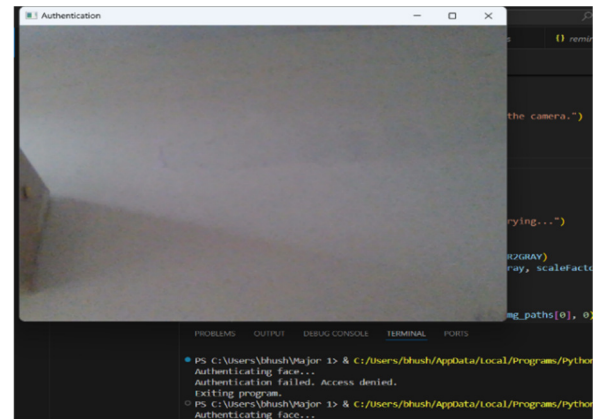


Fig. 8 Face Authentication Process

B. Voice and Text Command Interface:

The assistant accepts input through both voice and text as shown in figure 9 and figure 10. Speech recognition is handled via the browser using JavaScript Web Speech API. For users unable to speak or in noisy environments, a text input box is available. This dual-mode interface increases usability and accessibility.

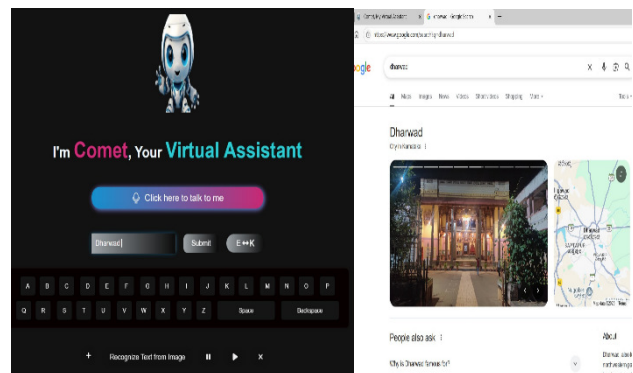


Fig. 9 Text Input and Search Output

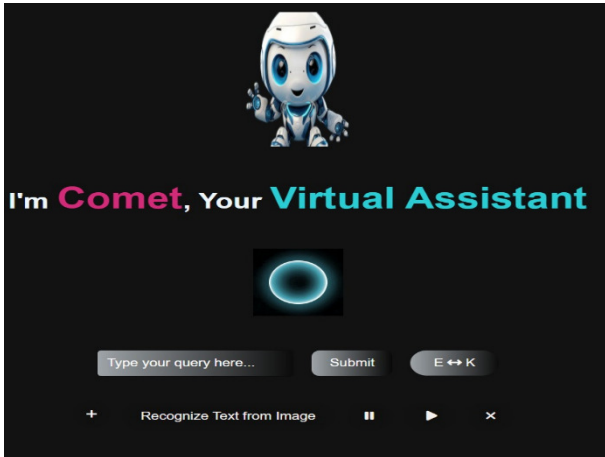


Fig. 10 Voice Input

C. Multi-Language Support:

To accommodate users who prefer different languages, the system supports both English and Kannada. The interface provides a language switch, and appropriate responses are synthesized using the browser’s speech Synthesis API. The Kannada input and its output is shown in figure 11.

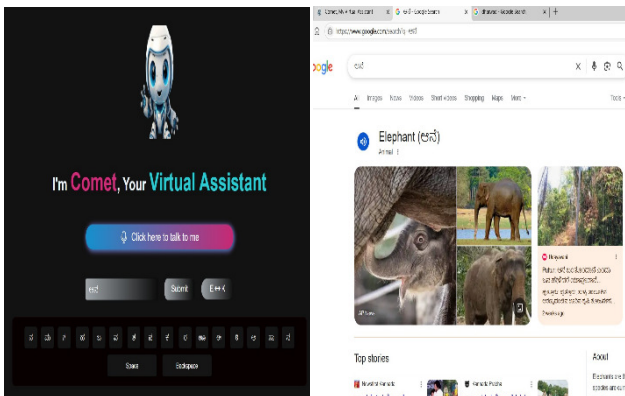


Fig. 11 Voice Input

D. OCR Integration:

The assistant is capable of reading text from images using Tesseract.js, a JavaScript-based OCR engine. Users can upload an image, and the assistant extracts the text and reads it aloud, making the tool useful for visually impaired individuals.

The text Extraction of English and Kannada from image are shown in figure 12 and figure 13.

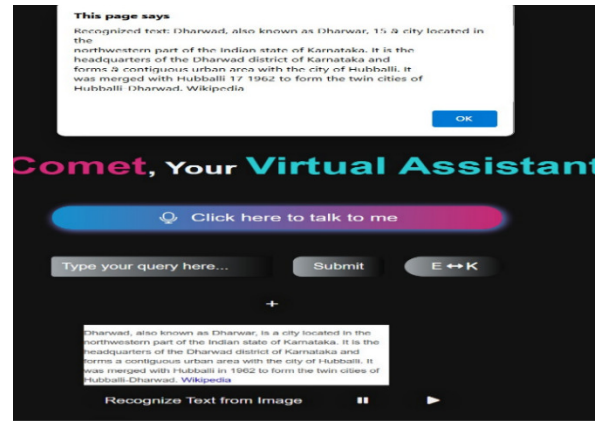


Fig. 12 Text Recognition from Image Features

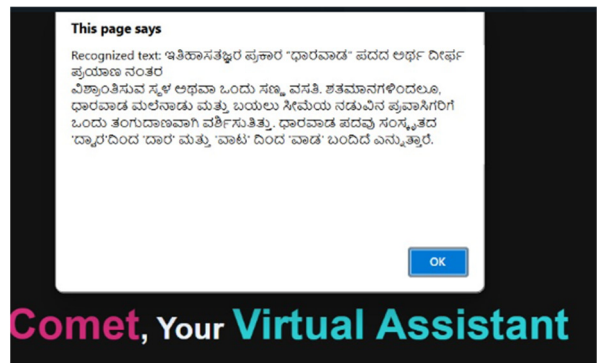


Fig. 13 Kannada Text Extraction

E. Help Section with Video Guide:

To assist new users, a dedicated Help section is provided, including a step-by-step video guide. This ensures that users of all technical backgrounds can understand and operate the assistant with ease. The Help Section with a Video Guide and Text are shown in figure 14.

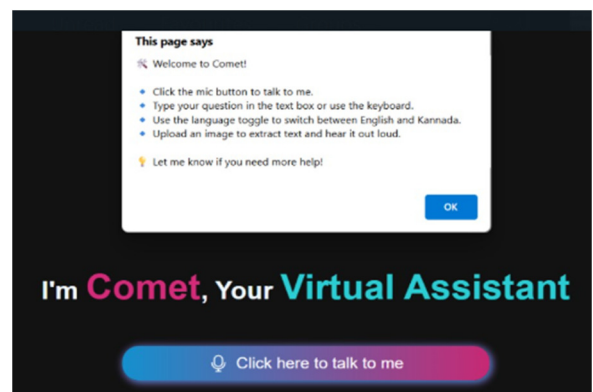


Fig. 14 Help Section with Text

F. Sidebar Product Search

An integrated sidebar enables users to search for products by name or keyword. It fetches and displays relevant search results, acting as a lightweight shopping assistant. The figure 15 shows the product search and its output.

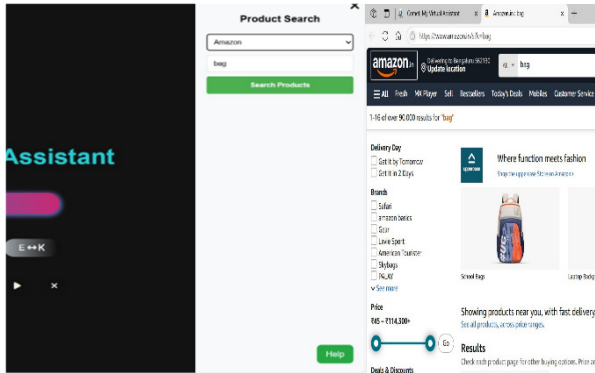


Fig. 15 Side bar for Product Search

VII. APPLICATIONS

Voice Assistants (VAs) are revolutionizing various industries by enabling intuitive and efficient user interactions. Here are some notable applications:

- **Medical Aid:** Voice assistants enhance e-health services, providing hands-free functionalities like blood pressure monitoring and patient care, which proved crucial during the COVID-19 pandemic. They contribute to improving healthcare accessibility while reducing costs [12].
- **Home Automation:** Modern home automation systems integrate voice inputs alongside text and email alerts, making homes smarter and more accessible. Future developments include regional language support to cater to diverse users, including individuals with physical challenges [13].
- **Education:** VAs is transforming education by offering natural language conversational interactions to enhance learning experiences. However, privacy and security concerns remain challenges to their widespread adoption in educational settings [14].
- **Automotive Industry:** In IoT-enabled smart cars, voice assistants improve the driving

experience by enabling voice-controlled operations such as moving, turning, or stopping, often powered by microcontrollers like ESP32 [15].

Voice assistants streamline tasks across these domains, offering hands-free convenience and promoting accessibility in everyday life.

VIII. CONCLUSIONS

The proposed system brings together face recognition and voice commands to create a secure and easy-to-use virtual assistant. It uses Python libraries like OpenCV, pyttsx3, and web browser along with JavaScript tools such as Speech Recognition, Speech Synthesis, and Tesseract.js to allow users to interact through voice, face login, and text extraction from images.

Features like a virtual keyboard, help video, product search, and support for both English and Kannada make the system more user-friendly and accessible. This project shows how combining vision, voice, and language can improve communication between humans and computers, especially for users who need hands-free or assistive support.

Since the system can work offline, it is also useful in places with limited internet. In the future, it can be improved by adding more languages, cloud storage, and AI-based personalized features for a better user experience.

REFERENCES

- [1] Aparna A.Patil, Ruchita Audumbar Gavali, Shabdali Suresh Shetty, "Voice Assistant," International Journal of Engineering Applied Sciences and Technology," 2021 Vol. 5, Issue 11, ISSN No. 2455-2143.
- [2] Ashim Jana, Dr. Harshali Patil, "The Evolution Of Voice Portal And Virtual Assistants," 2021 JETIR June 2021, Volume 8, Issue 6.
- [3] Baban Savic, Milos Milic, Sinisa Vlajic, "Analysis and Development of the Model for Google Assistant and Amazon Alexa Voice Assistants Integration," International Conference on Information Technology (IT) 2023.
- [4] Raj Kumar Jain, Vikas Sharma, Mangilal, Rakesh Kardam, Mamta Rani. "Artificial Intelligence Based A Communicative Virtual Voice Assistant Using Python & Visual Code Technology," World Journal of Research and Review (WJRR) ISSN: 2455-3956, Volume-13, Issue-5, November 2021.
- [5] Soumeek Mishra, Monali Nayak, "Artificially Intelligent Virtual Assistant Chatbot," International Journal of Research in Engineering and Science (IJRES) Volume 11 Issue 10 October 2023.

- [6] A. Sudhakar Reddy M, Vyshnavi, C. Raju Kumar, and Saumya ,“ Virtual Assistant using Artificial Intelligence,” JETIR March 2020, Volume 7, Issue 3.
- [7] Aishwarya C Maharajpet, Prof. Varsha S Jadhav, Ananya M Panchamukhi, Pranav Adagatti, Varshini. S. Gondkar, “Desktop Ai Assistant: J.A.R.V.I.S Just A Rather Very Intelligent System,” Journal of Emerging Technologies and Innovative Research, March 2024, Volume 11, Issue 3.
- [8] Rabin Joshi, Supriyo Kar, Abenezzer Wondimu Bamud and Mahesh T R, “Personal A.I. Desktop Assistant,” International Journal of Information Technology, Research and Applications 2023.
- [9] Prof. Rashmi Kannake, Pranmya Kale, Nikhil Dongre, Vipul Kshirsagar, Yogesh Tajane, “Smart Virtual Voice Assistant using python,” International Research Journal of Modernization in Engineering Technology and Science Volume:05/Issue:12/December-2023.
- [10] Himanshu Sharma, “Improving Natural Language Processing tasks by Using Machine Learning Techniques,” 2021 5th International Conference on Information Systems and Computer Networks (ISCON) GLA University, Mathura, India. Oct 22-23, 2021.
- [11] Vivek Vishal Singh, “ Virtual Assistant using Python,” Researchgate net publication May 2022.
- [12] Elaheh Ahanin, Abu Bakar Sade, Huam Hon Tat ,“ Applications of Artificial Intelligence and Voice Assistant in Healthcare”, . International Journal of Academic Research in Business and Social Sciences 19 December 2022.
- [13] Haris Isyanto, Ajib Setyo Arifin, Muhammad Suryanegara, “Design and Implementation of IoT-Based Smart Home Voice Commands for disabled people using Google Assistant,” International Conference on Smart Technology and Applications (ICoSTA) 2000.
- [14] Thanasis Tsourakas, George Terzopoulos and Stefanos Goumas, “Educational use of Voice Assistants and Smart Speakers,” Journal of Engineering Science and Technology 16 October 2021.
- [15] Yayshree Kumari, Raju Kumar Chaudhary, Rakesh Kumar, Meenu Gupta, “ IoT Based Voice Controlled Autonomous Robotic Vehicle Through Google Assistant,” 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N) 2021.