Journal of Computer and Electrical Sciences (JCES)- Volume-1 Issue-4, April 2026
ISSN: 3049-2602

AI-Powered Intelligent Desktop Assistant with Voice Authentication, Gesture Control, and Context-Aware Screen Interaction

Ch.Jakir Hussain Khan [1], U.Lokesh[2], M.Chenchi Reddy[3], Dr.Y.Prakash Rao[4]
[1,2,3,]Student, Department of Computer Science & Engineering
Andhra Loyola Institute of Engineering and Technology, ITI Road, Polytechnic Post, Vijayawada, Andhra Pradesh, India
[4]*Associate Professor, Department of Computer Science and Engineering*
Andhra Loyola Institute of Engineering and Technology, ITI Road, Polytechnic Post, Vijayawada, Andhra Pradesh, India
Email id: lokeshuppala88@gmail.com, machunuruchenchireddy0@gmail.com, jakirhussainkhan162@gmail.com,

Abstract: In recent years, intelligent virtual assistants have evolved beyond simple voice-based interaction to become **multimodal systems** capable of understanding speech, visual content, and human gestures. This project presents an advanced **desktop-based intelligent assistant** designed to provide natural, hands-free, and context-aware interaction between humans and computers, with a strong emphasis on enabling **accessible computing for individuals with physical and visual disabilities**.

The proposed system integrates **speech recognition, natural language understanding, computer vision, gesture recognition, and voice authentication** to deliver a seamless and secure user experience. Voice commands are processed using **real-time speech recognition**, enabling users to perform tasks such as opening applications, web browsing, YouTube playback, location searches, and retrieving date–time information without relying on traditional input devices. To enhance system security and personalization, a **voice authentication module** verifies the identity of the user based on unique vocal characteristics before granting access to sensitive operations or personalized features.

A conversational **AI-powered chatbot**, driven by large language models, provides intelligent responses, contextual assistance, and interactive communication with the user. To

https://doi.org/10.5281/zenodo.19600929

further improve accessibility, the system incorporates a **screen understanding module** that captures and interprets on-screen content using **Optical Character Recognition (OCR)** and vision- language models. This capability allows the assistant to verbally describe textual and visual information displayed on the screen, making it particularly beneficial for users with **visualimpairments**. When textual information is unavailable, **image-based scene understanding** is used to generate descriptive spoken explanations of visual content

Additionally, a **hotword detection mechanism** ensures the continuous availability of the assistant while minimizing computational overhead. A key feature of the system is its **gesture-controlled interface**, implemented using **MediaPipe and computer vision techniques**. Hand gestures are mapped to system-level operations such as cursor movement, clicking, scrolling, brightness adjustment, and volume control, enabling **touchless interaction for users with limited motor abilities**.

1. INTRODUCTION

In today's digital world, computer interaction primarily relies on traditional input devices such as keyboards and mice. However, this mode of interaction can be limiting for individuals with physical disabilities and may not provide a natural or intuitive user experience. The AI- Powered Intelligent Desktop Assistant is designed to transform human–computer interaction by integrating voice recognition, hand gesture control, screen understanding, and conversational AI into a single intelligent system. The system enables users to perform desktop operations, control applications, and retrieve information using voice commands and gestures, while also interpreting and describing on screen content for enhanced accessibility. This project aims to create a hands-free, context-aware, and inclusive computing environment using advanced AI and computer vision technologies. By leveraging real-time speech processing and computer vision, the assistant allows seamless multitasking without manual input devices. The integration of large language models enhances the assistant's ability to provide intelligent and contextual responses. Gesture-based interaction introduces a natural and touchless method of system control. Screen understanding capabilities improve accessibility for visually impaired users. Overall, the system represents a step toward next-generation smart and inclusive human computer interaction.

2.Literature Survey

1. Introduction

Recent advancements in Artificial Intelligence (AI), Natural Language Processing (NLP), and Computer Vision (CV) have significantly improved human–computer interaction. Intelligent systems can now process speech, text, and images simultaneously. However, most

https://doi.org/10.5281/zenodo.19600929

existing desktop assistants lack multimodal understanding and contextual awareness of screen content. This project builds upon prior research in voice assistants, OCR systems, gesture recognition, and large language models to develop an integrated AI-based desktop assistant.

2. Voice-Based Intelligent Assistants

Voice assistants such as Siri, Alexa, and Google Assistant use Automatic Speech Recognition (ASR), Natural Language Processing (NLP), and Text-to-Speech (TTS) technologies. These systems allow users to execute commands through speech. However, they primarily focus on predefined tasks and do not interpret on-screen visual content or provide contextual explanations of displayed information.

3. Screen Readers and Accessibility Tools

Screen readers like JAWS, NVDA, and Windows Narrator assist visually impaired users by reading structured interface elements. While effective for text-based applications, they depend on predefined UI metadata and cannot analyze images, charts, or complex graphical content. They also lack intelligent summarization capabilities.

4. Optical Character Recognition (OCR)

OCR technology, particularly Tesseract OCR, enables text extraction from images and screenshots. Modern OCR systems use deep learning-based models to improve recognition accuracy. However, OCR alone cannot interpret meaning or provide contextual explanations of extracted text, limiting its intelligence.

5. Computer Vision and Gesture Recognition

Vision-based interaction systems use frameworks like MediaPipe for real-time hand tracking and gesture recognition. These systems enable touchless control through landmark detection and image processing techniques. While effective for gesture-based interaction, they generally lack integration with intelligent reasoning systems.

6. Large Language Models (LLMs)

Transformer-based models such as GPT and LLaMA have significantly improved language understanding and text generation. These models can summarize, explain, and respond to user queries intelligently. However, traditional LLMs process only textual input and do not inherently understand visual content.

7. Vision-Language Models (VLMs)

Vision-Language Models integrate computer vision with language processing. Models like CLIP and LLaVA can describe images, identify objects, and answer visual questions. These systems provide contextual understanding of images but are often not integrated into desktop automation environments.

8. Research Gap

Existing systems focus on individual technologies such as voice control, OCR, or image

https://doi.org/10.5281/zenodo.19600929

recognition. There is limited integration of:

- Voice interaction
- Screen OCR
- Vision-based understanding
- Gesture control
- Large language models

Most current assistants cannot analyze and explain arbitrary screen content in real time.

9. Conclusion

Based on the literature review, it is evident that while significant progress has been made in AI, speech recognition, computer vision, and language modeling, fully integrated multimodal desktop assistants remain limited. The proposed system combines these technologies into a unified framework capable of understanding, interpreting, and explaining screen content using voice interaction and intelligent reasoning, thereby improving accessibility and human–computer interaction.

3. Proposed System

The proposed methodology follows a structured and modular approach to ensure flexibility and ease of implementation. The framework consists of data collection, preprocessing, model training, feature extraction, and decision support stages. Data in the form of audio signals for authentication and images for gesture tracking is captured in real time via microphone arrays and webcams. This ensures a transparent and responsive user experience.

Preprocessing steps include noise reduction for audio, feature scaling, and landmark normalization for hand gestures. These steps significantly improve model performance and interaction accuracy. Advanced machine learning models, specifically Gaussian Mixture Models (GMM) for voice biometric verification and Mediapipe neural networks for kinematic gesture detection, are selected based on their ability to capture complex behavioral signals while remaining computationally efficient.

Hybrid approaches that combine statistical and machine learning methods are used to improve stability and intent classification. The system is designed with a modular and scalable architecture that comprises speech-to-text translators, hotword listeners, executing agents utilizing large language models, and web automation tools. The architecture allows continuous deployment across desktop environments and readily adapts to noisy backgrounds using tailored acoustic thresholding without extensive modifications.

https://doi.org/10.5281/zenodo.19600929

After preprocessing, the models identify real-time command intent and execute localized system calls seamlessly. The trained system assists users with mobility or visual impairments to make better use of everyday operating system functions, online search, map routing, and hands-free messaging. The system also includes an interactive GUI frontend built using the Eel framework, providing dynamic visual feedback on status and transcription operations. The proposed system is designed to be simple, highly scalable, and suitable for real-world assistive computing applications.

4. Methodology

The proposed AI-Powered Intelligent Desktop Assistant follows a modular and multimodal architecture integrating voice authentication, speech processing, gesture recognition, and intelligent command execution. The system workflow is designed to ensure secure access, efficient interaction, and real-time response.

1. Voice Authentication Phase

The system begins with a voice-based authentication mechanism to ensure secure access. The user's voice is captured through a microphone and processed using Mel-Frequency Cepstral Coefficients (MFCC) feature extraction. These features are compared with pre-trained Gaussian Mixture Models (GMM) stored locally. Only if the voice matches the authorized profile, the system grants access to the assistant.

2. Voice Input Processing

Once authenticated, the system continuously listens for user commands. The Speech Engine captures audio input and converts it into text using Speech-to-Text (STT) techniques. This enables natural interaction between the user and the system.

3. Command Interpretation and Processing

The converted text command is passed to the Feature Manager, which acts as the central processing unit. It analyzes the command and determines the appropriate action, such as:

- Opening or closing applications
- Performing web searches or playing media
- Sending messages through integrated platforms
- Initiating chatbot-based conversations

4. System Execution Layer

Based on the interpreted command, the system interacts with:

- Operating System (OS) for local operations
- Web services for online tasks
- Database (SQLite) for retrieving stored commands and user data

Automation tasks are executed using libraries like PyAutoGUI and system-level

https://doi.org/10.5281/zenodo.19600929

commands.

5. Gesture Control Integration

The system also supports gesture-based interaction using a webcam. The Gesture Controller uses MediaPipe and OpenCV to track hand landmarks and map them to cursor movements and actions, enabling a virtual mouse interface for touchless control.

6. Context-Aware Screen Interaction

For enhanced intelligence, the system captures screen content and extracts text using Tesseract OCR. This allows the assistant to understand on-screen information and provide contextual responses.

7. AI-Based Response Generation

For complex queries, the system utilizes a Large Language Model (LLM) via API to generate intelligent responses. This enables natural conversations and improves user interaction quality.

8. Output Generation

Finally, the system provides feedback through:

- Text-to-Speech (TTS) for audio responses
- Graphical User Interface (GUI) using Eel for visual interaction

5. Proposed System Results

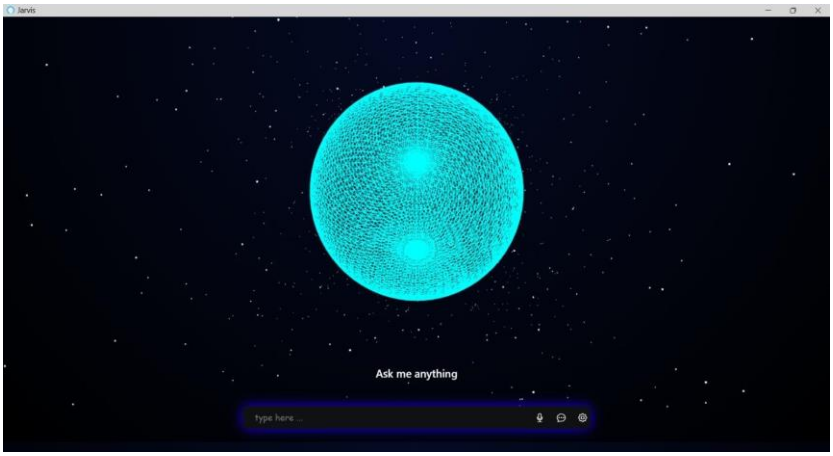


Fig:5.1 Interface of this project

a. Interface of the Project

The home interface of the **Jarvis Multimodal Voice Assistant System** presents a modern and interactive dashboard with a futuristic design. It displays the message **"Ask me anything"**, encouraging natural user interaction.

https://doi.org/10.5281/zenodo.19600929

The interface includes a text input field along with voice, chat, and settings options, enabling multiple modes of communication. Its simple and minimal design ensures easy navigation and quick access to features.

Key Features:

i. Voice and text-based interaction
ii. Real-time response display
iii. Interactive visual design
iv. User-friendly and accessible interface

This interface provides a seamless and efficient experience for users to interact with the system.

b. Response to the authorizer:

First, the system checks whether the user is authorized. If the user is authorized, it proceeds to the next step; otherwise, access is denied. This is the stage where authentication and

verification take place. The system ignores background noise and other disturbances, focusing solely on the authorized user's voice for processing. The system allows an authorized user to interact with the interface, providing all requested services seamlessly. By identifying and authenticating the user, the assistant proceeds to the next phase of service execution. Ultimately, the system remains fully active and responsive to user interactions, standing ready to provide services whenever a voice command is initiated.

FIG:- 5.2

FIG:- 5.3

https://doi.org/10.5281/zenodo.19600929

Fig 5.2. The user interface reacts as shown when a user calls Jarvis

Fig 5.3. This interface indicates that Jarvis is ready to listen to the authorized user..

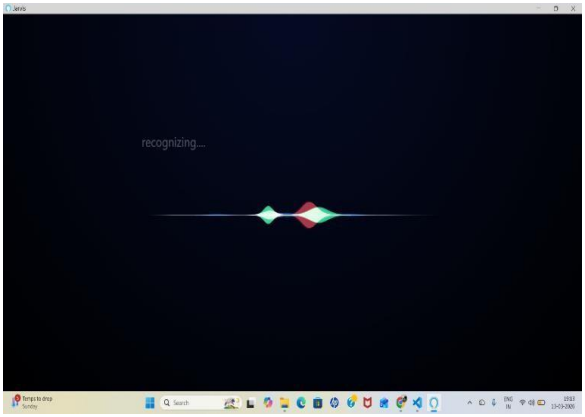


Fig:-5.4

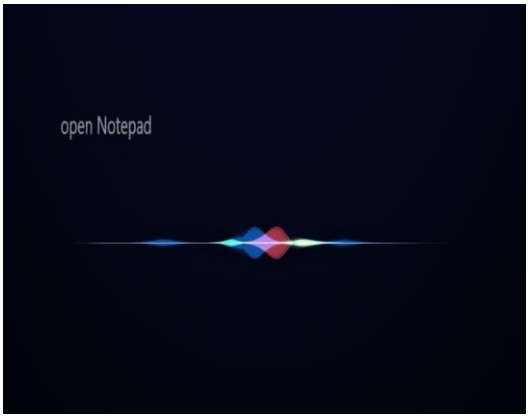


Fig:5.5

Fig 5.4:- Jarvis is ready to listen to the user's voice.

Fig 5.5 The system replies to the given question; the response is provided in a voice format. The assistant successfully launches the text editor and processes voice-to-text dictation for the user

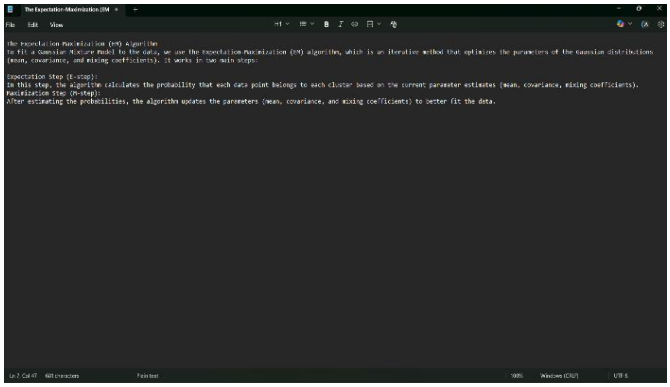


FIG:5.6

Fig 5.6- the output generated from the actions shown in Figures 5.4 and 5.5, specifically illustrating Jarvis opening the Notepad application. The system first verifies if the user is authorized; if they are, it responds and provides the requested services. Displays

6.AI chat Box

Jarvis is able to give answers for the given question and also replying in the form of voice to user.

https://doi.org/10.5281/zenodo.19600929

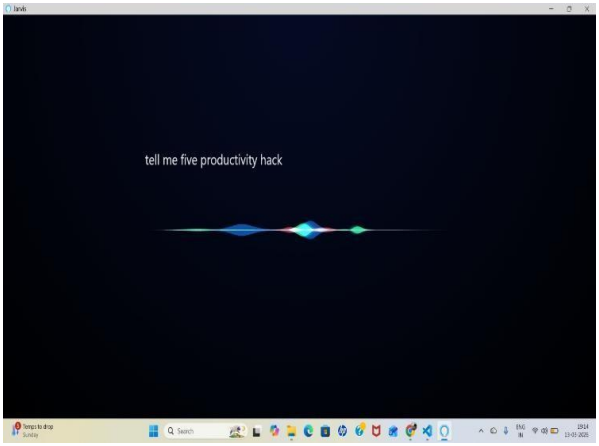


Fig 6.1

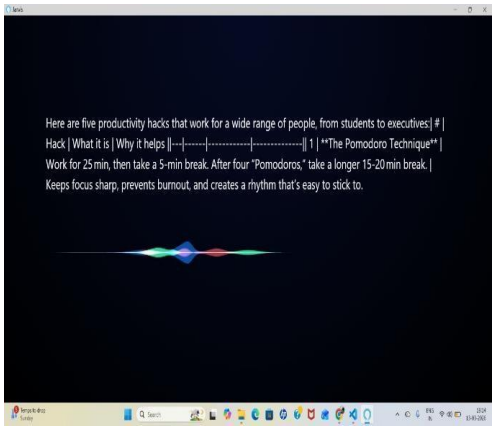


Fig 6.2

Fig 6.1- This illustrates the system receiving user input which doesn't belongs to predefined keyword.

Fig6.2- Response from LLM

A conversational **AI-powered chatbot**, driven by large language models, provides intelligent responses, contextual assistance, and interactive communication with the user.

Fig 6.3

Fig 6.3- The assistant is capable of saving the generated text, ensuring data persistence for future reference.

Screen Interaction: This capability allows the assistant to verbally describe textual and visual information displayed on the screen, making it particularly beneficial for users with **visual impairments**. When textual information is unavailable, **image-based scene understanding** is

https://doi.org/10.5281/zenodo.19600929

used to generate descriptive spoken explanations of visual content

Additionally, a **hotword detection mechanism** ensures the continuous availability of the assistant while minimizing computational overhead.

7.Gesture Control

A key feature of the system is its **gesture-controlled interface**, implemented using **MediaPipe and computer vision techniques**. Hand gestures are mapped to system-level operations such as cursor movement, clicking, scrolling, brightness adjustment, and volume control, enabling **touchless interaction for users with limited motor abilities**.

Fig7.1

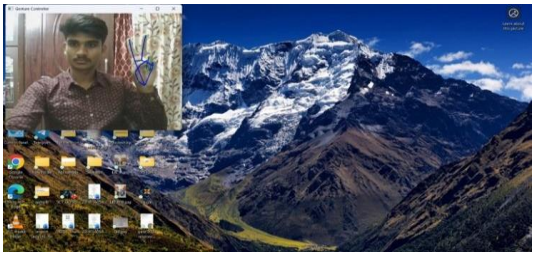


Fig7.2

Fig 7.1-This image indicates that the virtual mouse has been activated

Fig7.2- This represents the activation and opening of the Virtual Mouse interface.

The system also supports gesture-based interaction using a webcam. The Gesture Controller uses MediaPipe and OpenCV to track hand landmarks and map them to cursor movements and actions, enabling a virtual mouse interface for touchless control.

8 .Found Location:

Fig8.1;- Providing the input command: 'Find the location.

Fig 8.2:- Jarvis prompts the user to specify which location they would like to find

https://doi.org/10.5281/zenodo.19600929

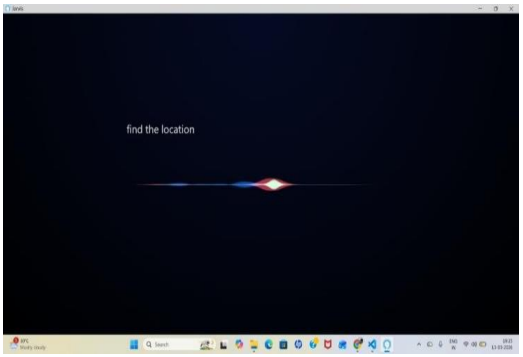


Fig 8.1

Fig 8.2

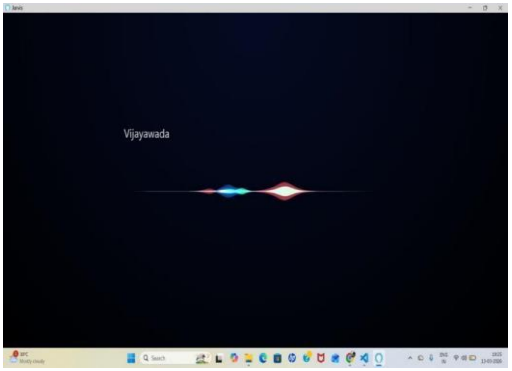


Fig 8.3

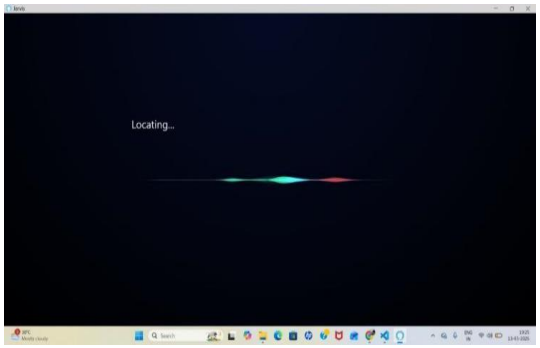


Fig 8.4

Fig 8.3:-Given the input to the Jarvis

Here giving the input Vijayawada and after going to analysing the input and taking voice only authorized user only and then it going to next step

Fig 8.4 :-Analysing the Input

The system processes the provided input, performs a search using Google Maps, and displays the resulting location

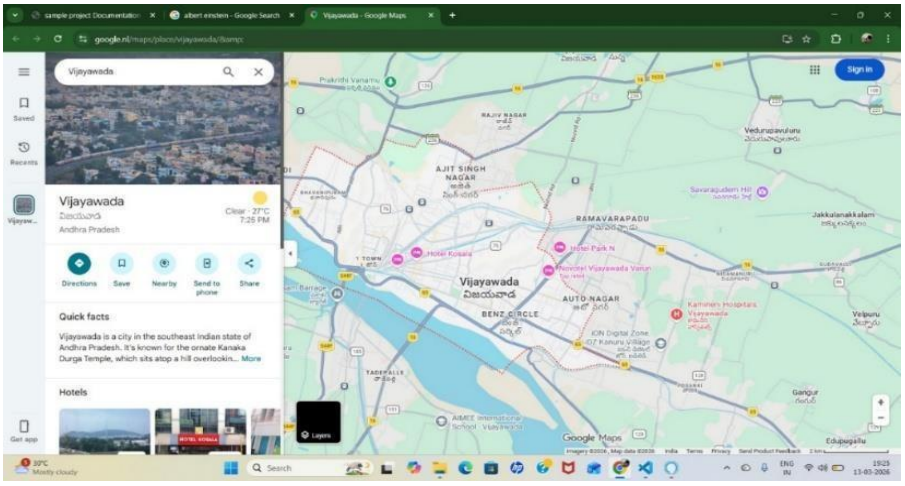


Fig 8.5:- Redirecting to the google map

https://doi.org/10.5281/zenodo.19600929

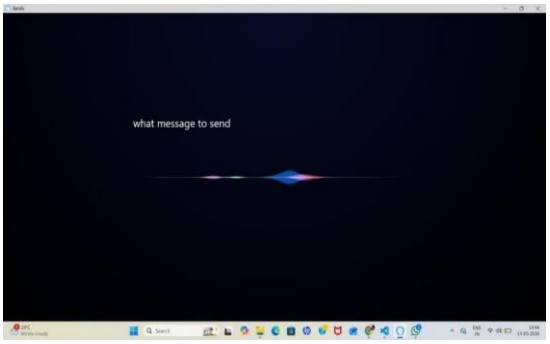


Fig 9.1 **Fig 9.2**

The system displays the location corresponding to the user's input, providing both directions and the distance from the current location to the destination. Operating as a virtual assistant, it provides these services to ensure a seamless navigation experience for the user

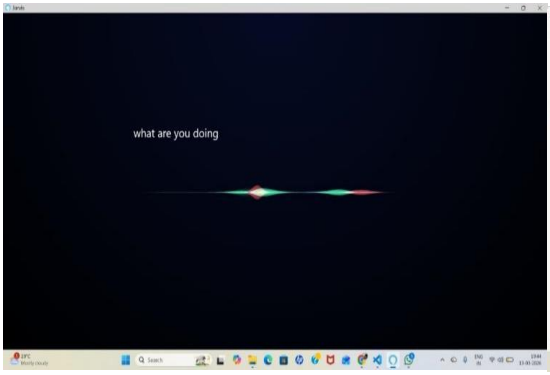
9.Message Transfer :

Jarvis is capable of sending messages to friends on WhatsApp by identifying and selecting names directly from the user's contact list

Fig 9.1:- Ask the message deliver to another person

The system sends messages to contacts in the WhatsApp list. These messages are processed and delivered in a voice-based format, allowing for hands-free communication.

Fig9.2:- Jarvis prompts the user to specify the message content that should be sent to the recipient.



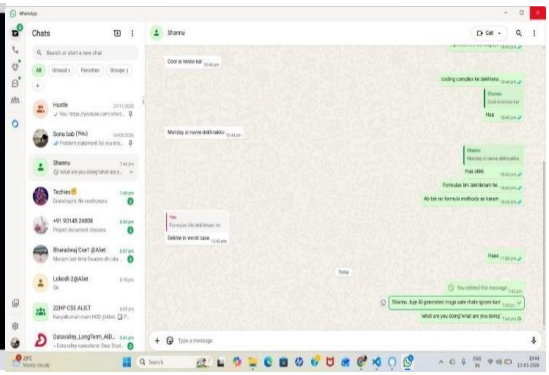


Fig 9.3 **Fig 9.4**

Fig 9.3:- Jarvis asks what message is going to be sent.

The assistant prompts the user for the content of the message to be sent to the receiver. This process functions similarly to manual instant messaging, but it is executed entirely through voice

https://doi.org/10.5281/zenodo.19600929

commands

Fig 9.4:- The results shown here represent the system's output during the execution of the Jarvis assistant.

The image above demonstrates a message being sent to 'Shanu.' Jarvis automatically identifies the recipient from the contact list and transmits the message via WhatsApp, mirroring the functionality of manual messaging through voice commands.

The **Message Transfer** module is a core feature of the system, designed to facilitate seamless communication through popular instant messaging platforms like **WhatsApp**. By leveraging voice recognition and automation scripts, the assistant allows users to send messages without any manual typing or physical interaction with a mobile device.

When a user initiates a messaging command, the system first triggers a search within the user's contact list. Jarvis utilizes string-matching algorithms to identify the correct recipient based on the voice input. Once a match is found, the assistant confirms the contact and prepares the communication channel.

After the recipient is identified, the system prompts the user to dictate the message. This voice input is converted into a text string using Speech-to-Text (STT) processing. As shown in Figure9.3, the assistant then automates the web or application interface to paste the content into the chat box and execute the "Send" command

10. Social Media Access

Jarvis is capable of providing both local and global news updates, functioning as a real-time information retrieval system

Fig10.1:- The assistant can access social media platforms to retrieve and display live news updates. By processing real-time feeds, Jarvis provides users with the latest information as events unfold

The system receives the voice prompt, '**Play Iran and Israel live on YouTube,**' and initiates the command processing phase. Jarvis analyzes the request, identifies the intent as a media search, and automatically retrieves the most relevant live news streams from YouTube.

https://doi.org/10.5281/zenodo.19600929

CONCLUSION

The AI-Powered Intelligent Desktop Assistant successfully integrates voice recognition, hand gesture control, screen understanding, and conversational AI into a unified system for natural human–computer interaction. By combining computer vision, speech processing, and large language models, the system enables complete hands-free control of desktop operations.The project demonstrates how multimodal AI can significantly improve accessibility, usability, and efficiency, especially for individuals with physical and visual disabilities.Overall, the system presents a scalable and intelligent framework for the future of smart and inclusive computing.

REFERENCES

[1] *Shaughnessy, IEEE, Interacting with Computers by Voice: Automatic Speech Recognition and Synthesis proceedings of the IEEE, vol. 91, no. 9, september 2003. [2] Patrick Nguyen, Georg Heigold, Geoffrey Zweig, Speech Recognition with Flat Direct Models, IEEE Journal of Selected Topics in Signal Processing, 2010. [3] Mackworth (2019-2020), Python code for voice assistant: Foundations of Computational Agents- David L. Poole and Alan, K. Mackworth.*

[2] *J, Prithvi, et al. 'Gesture Controlled Virtual Mouse with Voice Automation', www.ijert.org/research/gesture-controlled-virtual-mouse-with-v oice-automation-IJERTV12IS040131.pdf. Accessed 7 July 2023.*

[3] Jayasri Kotti1,∗ , B. Padmaja1 and D. Deepa2. *'Enhancing Gesture-Controlled Virtual Mouse and Virtual Keyboard Using AI Techniques'*

[4] Kavitha R1 , Janasruthi S U2 , Lokitha S3 , Tharani G4 , '*HAND GESTURE CONTROLLED VIRTUAL MOUSE USING ARTIFICAL INTELLIGENCE',*
http://ijariie.com/AdminUploadPdf/Hand_Gesture_Controlled_Virtual_Mouse_Using_Artificial
_Intelligence_ijariie19380.pdf

[5] GMM based automatic speaker verification system development for forensics in Bahasa Indonesia Ivan Stefanus;R.S. Joko Sarwono;Miranti Indar Mandasari 2017 5th International Conference on Instrumentation, Control, and Automation (ICA), 09-11 August 2017,
doi: 10.1109/ICA.2017.8068413

.

https://doi.org/10.5281/zenodo.19600929