IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, October 2026
Impact Factor: 8.2

A Low-Latency Wireless Smartphone-to-Desktop Controller with Real-Time Screen Mirroring, Motion, Touch, Voice and OCR

Samiksha Sachin Sharma[1], Aryan Anil Gaikwad[2], Chetan Prahlad Pardeshi[3], Shreyash Sanjay Bhavsar[4], Prof. Gokul Pawade[5]
Jagdamba Education Society's S. N. D. College of Engineering & Research Center, Yeola, Nashik

Abstract: *This paper presents the development of a low-latency wireless smartphone-to-desktop controller designed to improve the convenience, responsiveness, and accessibility of desktop interaction. The proposed system integrates smartphone-based motion control, touch gestures, virtual keyboard input, real-time screen mirroring, voice commands, and Optical Character Recognition (OCR) to provide an efficient alternative to traditional mouse and keyboard operations.*

The system uses a Flutter-based mobile application, a Spring Boot backend, and a Python desktop client. WebSocket communication with STOMP enables real-time, bidirectional data exchange between the smartphone and desktop. The smartphone's gyroscope and touchscreen are used to control cursor movement, clicking, and scrolling, while the desktop client executes the corresponding actions using PyAutoGUI. Screen mirroring is implemented by capturing desktop frames, compressing them into JPEG images, and transmitting them to the smartphone for real-time viewing.

Furthermore, the proposed system incorporates on-device speech recognition for hands-free desktop control and OCR for extracting text from the mirrored screen. A QR code or time-bound four-digit code is used for device pairing, helping establish session-specific communication between the connected devices. Processing voice commands and OCR directly on the smartphone can reduce dependence on cloud services, additional network requests, and recurring API costs.

This research focuses on the design and implementation of an integrated smartphone-based desktop control system, addressing challenges such as communication latency, cursor stability, screen-streaming performance, recognition accuracy, and secure device pairing. The proposed system aims to provide a portable, responsive, and user-friendly solution for presentations, teaching, office work, and remote desktop interaction. Performance evaluation will focus on latency, usability, connection reliability, and the accuracy of motion, voice, and OCR-based operations.

Keywords: Smartphone-Based Desktop Control, Wireless Communication, WebSocket, STOMP, Flutter, Spring Boot, Python, Motion Control, Screen Mirroring, Speech Recognition, Optical Character Recognition (OCR), Real-Time Interaction.

I. INTRODUCTION

In today's digital environment, desktop computers are widely used for presentations, teaching, office work, demonstrations, and various professional activities. However, traditional desktop interaction depends on physical input devices such as a mouse, keyboard, and presentation clicker. These devices may be inconvenient when users need to move away from their computers or control them from a distance. In this context, smartphone-based wireless control provides an opportunity to improve the convenience, portability, and accessibility of desktop interaction.

Smartphones contain several built-in capabilities, including gyroscopes, touchscreens, microphones, and cameras, that can be utilized to create an effective desktop control system. By integrating these features with wireless communication

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, October 2026
Impact Factor: 8.2

technologies, users can perform cursor movements, clicking, scrolling, and keyboard operations directly from their smartphones. Furthermore, real-time screen mirroring allows users to view the desktop display on their mobile devices, providing visual feedback while controlling the computer remotely.

This paper focuses on the design and development of a low-latency wireless smartphone-to-desktop controller that integrates motion control, touch gestures, virtual keyboard input, screen mirroring, voice commands, and Optical Character Recognition (OCR). The proposed system uses a Flutter-based mobile application, a Spring Boot backend, and a Python desktop client connected through WebSocket communication with STOMP. Gyroscope-based cursor movement and touch gestures enable natural interaction, while screen capture and image compression facilitate the transmission of desktop frames to the smartphone. On-device speech recognition and OCR provide additional functionality while reducing dependence on external cloud services.

The research explores the architecture, implementation, and practical applications of the proposed system, with particular attention to communication latency, cursor stability, screen-streaming performance, recognition accuracy, and secure device pairing. By addressing these challenges, the proposed work aims to provide a portable, responsive, and user-friendly alternative to conventional desktop input devices for presentations, educational activities, office tasks, and remote computer interaction.

Sign Language Translator:

In today's digital environment, controlling desktop computers remotely can be challenging when users depend entirely on traditional input devices such as a mouse, keyboard, and presentation clicker. These devices may limit mobility and convenience during presentations, educational activities, office work, and demonstrations. Existing remote-control solutions may also introduce communication delays or lack integrated features for motion-based control, screen mirroring, voice commands, and text recognition.

The proposed Smartphone-Based Wireless Desktop Controller addresses these challenges by integrating smartphone sensors, wireless communication, and artificial intelligence-based features to enable convenient desktop interaction. The system utilizes the smartphone's gyroscope, touchscreen, microphone, and camera to perform cursor movements, clicking, scrolling, and keyboard operations. It also supports real-time screen mirroring, allowing users to view the desktop display directly on their smartphones while controlling the computer.

This solution integrates a Flutter-based mobile application, a Spring Boot backend, and a Python desktop client using WebSocket/STOMP communication. On-device speech recognition enables hands-free desktop operations, while Optical Character Recognition (OCR) extracts text from the mirrored screen. Secure device pairing through a QR code or time-bound four-digit code helps establish communication between the intended smartphone and desktop. By combining these functionalities into a single system, the proposed solution aims to improve portability, responsiveness, and usability while reducing dependence on additional hardware and external cloud services.

AI-Powered Desktop Control and Real-Time Screen Mirroring:

Traditional desktop control relies on physical input devices such as a mouse, keyboard, and presentation clicker. These devices may be inconvenient during presentations, teaching, and office work, particularly when users need to control a computer from a distance. The proposed system addresses this challenge by transforming a smartphone into a wireless desktop controller using its built-in gyroscope, touchscreen, microphone, and camera.

The system uses a Flutter-based mobile application, a Spring Boot backend, and a Python desktop client to provide wireless desktop interaction. WebSocket with STOMP enables real-time communication between the smartphone and desktop. Gyroscope-based motion control and touch gestures support cursor movement, clicking, and scrolling, while screen capture using MSS and JPEG compression enables desktop screen mirroring on the smartphone.

The system also incorporates on-device speech recognition for voice commands and Optical Character Recognition (OCR) to extract text from the mirrored desktop screen. QR-code or time-bound four-digit-code pairing establishes a

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, October 2026
Impact Factor: 8.2

connection between the intended devices. These features aim to improve portability, responsiveness, and usability while reducing dependence on additional hardware and external cloud services.

Algorithms and Techniques Used in the Proposed System:

- **Motion Control Algorithm –** Uses smartphone gyroscope data to control desktop cursor movement. Threshold filtering helps reduce unwanted jitter and provides smoother cursor movement.
- **Gesture Recognition** – Detects touch gestures such as tap, hold, double-tap, and swipe, mapping them to desktop actions such as clicking and scrolling.
- **Voice Command Processing** – Uses the smartphone's native, on-device speech-to-text engine to recognize commands and convert them into desktop actions, including clicking, scrolling, typing, and keyboard navigation.
- **Optical Character Recognition (OCR) –** Uses an on-device text-recognition model to extract text from mirrored desktop frames and identify text blocks with their bounding boxes.
- **Screen Streaming Algorithm** – Captures the desktop screen using MSS, resizes and compresses frames into JPEG images, and transmits them to the smartphone. Adaptive image quality and frame-rate adjustments help manage network performance.
- **WebSocket/STOMP Communication** – Enables bidirectional communication between the Flutter mobile application, Spring Boot backend, and Python desktop client, supporting the exchange of control commands and screen frames.

II. PROBLEM STATEMENT

Traditional desktop control depends on physical devices such as a mouse, keyboard, and presentation clicker, which can limit convenience and mobility during presentations, teaching, and office work. Existing remote-control solutions may not provide an integrated experience for motion-based cursor control, touch gestures, virtual keyboard input, and real-time screen mirroring. Furthermore, communication delays and dependence on cloud-based services can affect responsiveness and accessibility. The proposed smartphone-based wireless desktop controller aims to address these challenges by integrating smartphone gyroscope-based motion control, touchscreen gestures, voice commands, and optical character recognition (ocr). Using flutter, spring boot, websocket/stomp, and python, the system enables wireless desktop interaction, real-time screen feedback, and on-device voice and text recognition, providing a portable, responsive, and convenient alternative to traditional desktop input devices.

III. OBJECTIVES

I. Develop a low-latency wireless smartphone-to-desktop controller for convenient remote computer interaction.
II. Implement gyroscope-based motion control and touch gesture recognition to support cursor movement, clicking, double-clicking, and scrolling.
III. Integrate real-time desktop screen mirroring using screen capture, jpeg compression, and websocket/stomp communication.
IV. Implement a virtual keyboard and on-device voice command processing to enable remote typing and hands-free desktop control.
V. Integrate on-device optical character recognition (ocr) to extract text from the mirrored desktop screen.
VI. Establish secure smartphone–desktop pairing using a qr code or time-bound four-digit code.
VII. Improve system responsiveness, usability, and reliability by evaluating communication latency, motion accuracy, screen-streaming performance, and connection stability.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, October 2026
Impact Factor: 8.2

Real-time monitoring and performance optimization

The proposed system supports real-time monitoring of communication and desktop interaction to improve responsiveness and reliability.

- real-time monitoring of websocket/stomp communication between the smartphone and desktop.
- continuous screen capture and streaming with adaptive jpeg quality and frame-rate adjustments.
- motion-data filtering to reduce gyroscope jitter and provide smoother cursor movement.
- performance evaluation of communication latency, screen-mirroring stability, and connection reliability.
- testing voice-command recognition and ocr accuracy to assess the effectiveness of on-device processing.

Significance of the study:

This research addresses the need for convenient, portable, and responsive desktop control by utilizing the capabilities of modern smartphones. Traditional input devices such as mice, keyboards, and presentation clickers may limit mobility and convenience during presentations, teaching, and office work. The proposed smartphone-based wireless desktop controller aims to simplify computer interaction through motion control, touch gestures, virtual keyboard input, and real-time screen mirroring.

By integrating flutter, spring boot, websocket/stomp communication, and python, the study explores an efficient approach to wireless desktop interaction. The inclusion of on-device voice recognition and optical character recognition (ocr) provides additional functionality for hands-free control and text extraction. Secure device pairing and adaptive screen streaming further contribute to the system's usability and reliability.

The findings of this research may contribute to the development of practical human-computer interaction solutions for educational institutions, professional environments, presentations, and demonstrations. By evaluating communication latency, cursor responsiveness, screen-streaming performance, and recognition accuracy, the study aims to identify opportunities for improving accessibility, portability, and overall user experience while reducing dependence on additional hardware and external cloud services.

IV. PURPOSED METHODOLOGY

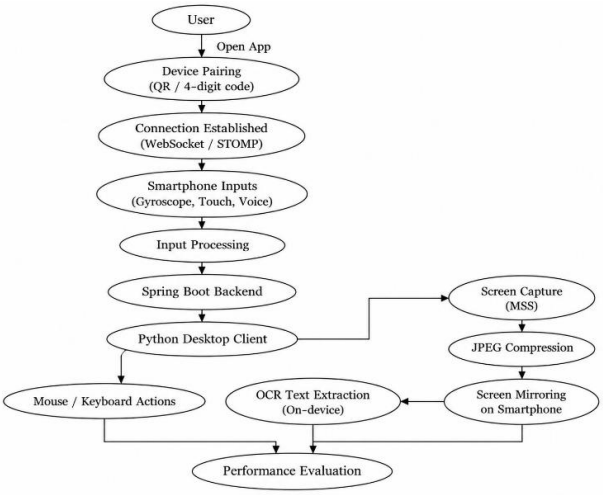


Fig. 1 Workflow Diagram

- **Proposed Methodology for Smartphone-Based Wireless Desktop Controller**

This research follows a system development and performance evaluation approach to design and implement a low-latency wireless smartphone-to-desktop controller. The methodology focuses on integrating motion control, touch gestures, virtual keyboard input, real-time screen mirroring, voice commands, and Optical Character Recognition

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 7, October 2026 Impact Factor: 8.2

(OCR) into a unified system. The proposed architecture consists of a Flutter mobile application, a Spring Boot backend, and a Python desktop client connected through WebSocket/STOMP communication.

The system is developed and tested through individual modules, including secure device pairing, gyroscope-based cursor movement, gesture recognition, desktop command execution, screen capture, and on-device voice and text recognition. Quantitative performance evaluation focuses on communication latency, cursor responsiveness, screen-streaming performance, voice-command recognition, OCR accuracy, and connection reliability. Functional testing is also conducted to assess usability, pairing, reconnection, and system behaviour under different network conditions. The results will help identify performance limitations and guide optimization of the proposed desktop control system.

- **DATA COLLECTION & STUDY DESIGN**
- **1. QUANTITATIVE ANALYSIS:**
- **System Performance:** The proposed system will be evaluated based on communication latency, cursor responsiveness, screen-streaming performance, voice-command recognition, and OCR accuracy.
- **Connection Reliability:** Testing will assess successful device pairing, reconnection behavior, and stable communication between the smartphone and desktop.
- **System Response Time:** Measurement of the time required to transmit user inputs, execute desktop actions, and deliver screen frames to the smartphone.

V. FUNCTIONAL ANALYSIS

Module Testing: Individual modules, including device pairing, motion control, touch gestures, virtual keyboard input, screen mirroring, voice commands, and OCR, will be tested.

Observational Testing: The system will be tested using a smartphone and desktop connected over the same network to evaluate real-time interaction and usability.

System Implementation and Processing:

Device Pairing & Communication: The smartphone and desktop are connected using a QR code or a time-bound four-digit code, with WebSocket/STOMP supporting real-time communication.

Motion & Touch Processing: Gyroscope data is filtered to reduce unwanted jitter, while touch gestures are mapped to desktop actions such as clicking and scrolling.

Desktop Control: The Python desktop client receives commands and executes mouse and keyboard actions using PyAutoGUI.

Screen Capture & Mirroring: MSS captures the desktop screen, JPEG compression reduces frame size, and compressed frames are transmitted to the smartphone.

Voice Command Processing: The smartphone's native on-device speech-to-text engine recognizes supported commands and forwards them to the desktop client.

OCR Processing: An on-device text-recognition model extracts text blocks and bounding boxes from the mirrored desktop screen.

Performance Evaluation: Testing focuses on latency, responsiveness, streaming quality, recognition accuracy, and connection reliability.

Key Frameworks And Technologies Used: Flutter, Dart, Spring Boot, Java 17, Websocket/Stomp, Python, Pyautogui, Mss, Pillow, Qrcode, Ml Kit, And Native Speech-To-Text.

- **EVALUATION METRICS**
- **QUANTITATIVE METRICS:**

Communication latency between the smartphone, Spring Boot backend, and desktop client.

Cursor responsiveness and accuracy of motion-based control.

Screen-streaming performance, including frame rate, image quality, and transmission stability.

Voice-command recognition accuracy for supported desktop operations.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 7, October 2026 Impact Factor: 8.2

OCR accuracy in extracting text from mirrored desktop screens.
Connection reliability, including successful device pairing and reconnection performance.

- **QUALITATIVE METRICS:**

User satisfaction with the smartphone-based desktop control system.
Ease of use of touch gestures, virtual keyboard input, and voice commands.
Perceived usefulness of real-time screen mirroring during presentations, teaching, and office work.
Overall usability and reliability of the integrated system.

- **ETHICAL CONSIDERATIONS**

Data Privacy: Protect screen data and user inputs through secure communication and device pairing.
Access Control: Allow connections only between authorized devices.
On-Device Processing: Reduce cloud dependency for voice recognition and OCR.

EXPECTED OUTCOMES

Wireless desktop control using a smartphone.
Real-time screen mirroring with responsive cursor movement.
Voice commands and ocr-based text extraction.
Improved usability, reliability, and reduced cloud dependency.

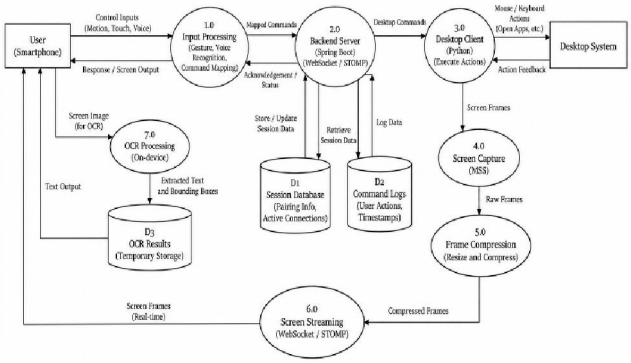


Fig. 2 Data Flow Diagram

Data Collection (Motion, Touch, and Voice Inputs)
→ Raw Input Data
Data Preprocessing (Noise Filtering, Gesture Recognition, Command Mapping)
→ Processed Input Data
→ Send Commands to Backend
Backend Processing (Spring Boot, WebSocket/STOMP)
→ Forward Commands to Python Desktop Client
→ Manage Session and Connection Data
Desktop Control (PyAutoGUI)
→ Mouse and Keyboard Actions
Screen Capture (MSS)
→ Raw Screen Frames
Frame Processing (Resizing and JPEG Compression)
→ Compressed Screen Frames

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 7, October 2026 Impact Factor: 8.2

→ WebSocket/STOMP Streaming to Smartphone
Screen Mirroring
→ Real-Time Desktop Screen Display
OCR Processing (On-Device Text Recognition)
→ Extracted Text and Bounding Boxes

VI. THEORETICAL FRAMEWORK

The theoretical framework for the smartphone-based wireless desktop controller integrates concepts from human-computer interaction, wireless communication, real-time data processing, and assistive technology. These concepts guide the development of a system that enables smartphone-based desktop control through motion, touch, voice commands, screen mirroring, and Optical Character Recognition (OCR).

- **Human-Computer Interaction (HCI):**
  The HCI framework focuses on improving the interaction between users and desktop computers. The proposed system uses smartphone gyroscope data, touch gestures, and a virtual keyboard to perform desktop operations, providing a convenient alternative to traditional mouse and keyboard input.
- **Real-Time Communication Framework:**
  The system uses WebSocket/STOMP communication through a Spring Boot backend to exchange control commands and screen frames between the smartphone and desktop client. This supports responsive interaction and real-time screen mirroring.
- **Motion and Gesture Recognition:**
  The system processes gyroscope data and touchscreen gestures to control cursor movement, clicking, and scrolling. Motion filtering helps reduce unwanted cursor jitter and improves the smoothness of desktop interaction.
- **On-Device Processing Framework:**
  The system incorporates on-device speech recognition and Optical Character Recognition (OCR). Voice commands enable hands-free desktop operations, while OCR extracts text from mirrored desktop screens, reducing dependence on external cloud services.
- **Synthesis of Frameworks:**
  The integration of human-computer interaction, real-time communication, motion processing, and on-device recognition provides the foundation for the proposed smartphone-to-desktop controller. These concepts guide the system's design and implementation, with performance evaluation focusing on communication latency, cursor responsiveness, streaming quality, recognition accuracy, and connection reliability.

VII. CONCLUSION: SMARTPHONE-BASED WIRELESS DESKTOP CONTROLLER

The research presented in this paper focuses on the development of a low-latency wireless smartphone-to-desktop controller designed to improve the convenience, responsiveness, and accessibility of desktop interaction. By integrating motion control, touch gestures, virtual keyboard input, voice commands, real-time screen mirroring, and Optical Character Recognition (OCR), the proposed system provides an alternative to traditional mouse and keyboard operations.

The proposed architecture combines a Flutter mobile application, a Spring Boot backend, and a Python desktop client using WebSocket/STOMP communication. Gyroscope-based cursor movement and touch gestures support desktop operations, while MSS screen capture and JPEG compression facilitate screen streaming. On-device speech recognition and OCR provide additional functionality while reducing dependence on external cloud services.

The system aims to support presentations, teaching, office work, and remote desktop interaction through a unified smartphone-based interface. Its performance is evaluated in terms of communication latency, cursor responsiveness,

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 7, October 2026 Impact Factor: 8.2

screen-streaming quality, voice and OCR accuracy, and connection reliability. These evaluation criteria provide a foundation for identifying limitations and optimizing the overall system.

Future improvements may focus on enhancing motion stability, reducing communication delays, optimizing screen-streaming performance, strengthening device-pairing security, and improving voice-command and OCR accuracy. Overall, the proposed system demonstrates the potential of integrating smartphone sensors, wireless communication, and on-device processing to create a portable and user-friendly desktop control solution.

Future Scope: Advancements in Smartphone-Based Wireless Desktop Control

A. Expanding Control Capabilities

Future versions could incorporate additional smartphone sensors and advanced gesture recognition to improve cursor movement, scrolling, and desktop navigation. Improved voice-command processing could support a wider range of desktop operations.

B. Cross-Platform Compatibility

The system could be extended to support multiple operating systems, including Windows, Linux, and macOS. Compatibility with tablets and other smart devices could further expand its applications.

C. Enhanced Performance and Optimization

Future development could focus on reducing communication latency, improving cursor stability, and optimizing JPEG compression and screen-streaming frame rates for smoother real-time interaction.

D. Advanced Voice and OCR Features

Voice recognition could be enhanced to support additional languages and more complex commands. OCR could be improved to extract text more accurately from different screen layouts and document formats.

E. Improved Security and Device Pairing

Future versions could introduce stronger authentication, encrypted communication, session expiration, and improved access control to strengthen the security of smartphone-to-desktop connections.

F. Integration with Smart Devices

The system could be extended to control presentations, media playback, and other authorized desktop applications through a unified smartphone interface, improving its usefulness in educational and professional environments.

G. Further Testing and Evaluation

Future research could evaluate the system across different devices, network conditions, and desktop configurations. Testing could focus on latency, connection reliability, screen-streaming quality, voice-command recognition, OCR accuracy, and overall user experience to guide further improvements.

Collaborating with software developers, academic institutions, and technology experts could help validate the proposed smartphone-to-desktop controller, ensuring reliable communication, secure device pairing, and effective desktop interaction. Testing feedback would play a crucial role in improving cursor responsiveness, screen-mirroring performance, voice-command recognition, and OCR accuracy while enhancing the overall usability and reliability of the system.

Future research could focus on testing the system across different operating systems, smartphones, desktop configurations, and network conditions. Performance evaluation would help identify communication delays, screen-streaming limitations, and connectivity issues. Improvements in motion filtering and gesture recognition could provide smoother and more accurate cursor control.

Further development could enhance on-device speech recognition and OCR capabilities to support more commands and improve text extraction from mirrored screens. Security improvements, including stronger authentication and secure session management, could help protect communication between paired devices. Adaptive image compression and frame-rate optimization could also improve screen mirroring under limited network bandwidth.

The system could eventually be extended with additional desktop shortcuts, multimedia controls, and presentation features to support educational institutions, offices, and professional demonstrations. Continuous testing, user feedback,

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, October 2026
Impact Factor: 8.2

and technical improvements would help establish a more responsive, secure, accessible, and reliable smartphone-based desktop control solution.

REFERENCES

1. A. Author(s), "Smartphone-Enabled Interaction on Large Displays—A Web-Technology-Based Approach," Electronics, vol. 13, no. 5, article 929, 2024. https://doi.org/10.3390/electronics13050929
2. Z. Qiu, S. Yang, Y. Yu, Y. Luo, and W. Diao, "Understanding Security Risks in Mobile-to-PC Screen Mirroring: An Empirical Study," Cybersecurity, vol. 8, article 68, 2025. https://doi.org/10.1186/s42400-025-00363-8.
3. S. Mahalakshmi, N. Baskaran, R. Hemamalini, K. Kowsalyadevi, and P. Partheeban, "Android Espion: Integrated Remote Control for Android Device," in Proceedings of the IEEE 9th International Conference on Information and Communication Technology (CICT), 2025. https://doi.org/10.1109/CICT67193.2025.11399038
4. M. G. Gonzales, P. Corcoran, N. Harte, and M. Schukat, "On-Device Evaluation of Joint Speech-Text Embeddings for ASR, TTS, and Speaker Recognition Tasks," IEEE Access, vol. 13, pp. 192518–192530, 2025. https://doi.org/10.1109/ACCESS.2025.3630503
5. W. S. Hoh, N. S. I. Mohamad Azizi, B. L. Ong, S. K. Yoon, and M. QiYuan, "ScanNote: A Mobile Application for Enhanced Text Recognition and Digital Note-Taking Using Machine Learning-Driven Optical Character Recognition (OCR)," Journal of Advanced Research Design, vol. 124, no. 1, pp. 77–89, 2025. https://doi.org/10.37934/ard.124.1.7789
6. A. Poudel, A. Parajuli, B. Khanal, N. Danai, and R. Nepal, "A Privacy-Focused Cross-Platform Peer-to-Peer Communication and Synchronization Framework Using WebRTC and WebSocket," Proceedings of the International Conference on Innovation in Computing, Science, Engineering and Technology (ICICSET), vol. 3, no. 1, 2026.
7. Z. Qiu, S. Yang, Y. Yu, Y. Luo, and W. Diao, "Understanding Security Risks in Mobile-to-PC Screen Mirroring: An Empirical Study," Cybersecurity, vol. 8, article 68, 2025. https://doi.org/10.1186/s42400-025-00363-8
8. S. A. Macdonald, M. S. Islam, A. Bodrogai, J. Finlow, F. Colquhoun, J. H. Williamson, and M. Khamis, "SafeScreen: Evaluating a Screen-Detecting Smartphone Camera App Under Benign and Adversarial Use," Proceedings of the ACM on Human-Computer Interaction, vol. 9, no. 5, article MHCI025, 2025. https://doi.org/10.1145/3743715.
9. N. Shah, B. Leung, M. Shimabukuro, and A. Neshati, "SwipeSense: Exploring the Feasibility of Back-of-Device Swipe Interaction Using Built-In IMU Sensors," Proceedings of the ACM on Human-Computer Interaction, vol. 9, no. 5, article MHCI030, 2025. https://doi.org/10.1145/3743734
10. S. Mahalakshmi, N. Baskaran, R. Hemamalini, K. Kowsalyadevi, and P. Partheeban, "Android Espion: Integrated Remote Control for Android Device," in Proceedings of the IEEE 9th International Conference on Information and Communication Technology (CICT), 2025. https://doi.org/10.1109/CICT67193.2025.11399038.
11. P. Mohana Priya, J. Meyyarivu, and S. Pradeesh, "A Real-Time Multimodal AI Assistant Using Hand Gestures and Speech Together," in Proceedings of the 8th International Conference on Recent Advances in Science, Engineering and Management, Chennai, 2026. Available at: https://ir.vistas.ac.in/id/eprint/18489/
12. "NEXORA: Eye Tracking, Voice Command and Gesture Integration System for Enhanced Human-Computer Interface," 2026 International Conference on Recent Advancement in Electrical, Computer and Communication Technologies (IECCT), 2026. DOI: https://doi.org/10.1109/IECCT68664.2026.11542046

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, October 2026
Impact Factor: 8.2

13. S. Jaya Naga Chandrika, V. Swapna, K. Revathi, K. Ratna Kumar, and G. Hanok, "A Multimodal Gesture and Voice Controlled Virtual Mouse for Accessible Desktop Interaction," 2026. https://doi.org/10.22214/ijraset.2026.83951
14. G. V. S. Santhosh, A. Yaswanth, B. Vamsi, B. Vijay Teja, and M. Hanumantha Rao, "Intelligent Desktop Interface: A Multimodal Human–Computer Interaction System Integrating Hand Gesture Recognition and Voice Command Processing," International Journal of Versatile Research and Analysis, 2026.
15. A. Ahirrao, S. Shinde, B. S. Virdee, and A. Khanna, "An Intelligent Mouse Control Interface Using Multimodal Interaction," IEEE conference paper, 2026. https://doi.org/10.1109/ICICIS65613.2025.11371134
16. P. Mohana Priya, J. Meyyarivu, and S. Pradeesh, "A Real-Time Multimodal AI Assistant Using Hand Gestures and Speech Together," in Proceedings of the 8th International Conference on Recent Advances in Science, Engineering and Management, Chennai, 2026. https://ir.vistas.ac.in/id/eprint/18489/
17. "Real-Time Gesture Recognition on Mobile Devices Using Deep Learning with Fully on-Device Inference," in Proceedings of the 5th International Conference on Innovative Practices in Technology and Management (ICIPTM), 2026. https://doi.org/10.1109/ICIPTM69057.2026.11466037.
18. "NEXORA: Eye Tracking, Voice Command and Gesture Integration System for Enhanced Human-Computer Interface," in Proceedings of the International Conference on Recent Advancement in Electrical, Computer and Communication Technologies (IECCT), 2026. https://doi.org/10.1109/IECCT68664.2026.11542046
19. G. V. S. Santhosh, A. Yaswanth, B. Vamsi, B. Vijay Teja, and M. Hanumantha Rao, "Intelligent Desktop Interface: A Multimodal Human–Computer Interaction System Integrating Hand Gesture Recognition and Voice Command Processing," International Journal of Versatile Research and Analysis, 2026.
20. S. Jaya Naga Chandrika, V. Swapna, K. Revathi, K. Ratna Kumar, and G. Hanok, "A Multimodal Gesture and Voice Controlled Virtual Mouse for Accessible Desktop Interaction," 2026. https://doi.org/10.22214/ijraset.2026.83951
21. "EchoSign: A Mobile Application for Sign Language Recognition and Translation," in Proceedings of the 5th International Conference on Innovative Computing (ICIC), 2024. https://doi.org/10.1109/ICIC63915.2024.11116422
22. "Understanding Security Risks in Mobile-to-PC Screen Mirroring: An Empirical Study," Cybersecurity, vol. 8, article 68, 2025. https://doi.org/10.1186/s42400-025-00363-8.