ENHANCING USER EXPERIENCE WITH HAND GESTURE VOLUME CONTROL: OVERCOMING CHALLENGES OF CAMERA DISTANCE

Shamil Kh.Ramadhan
Department of Medical Instrumentation Techniques
Northern Technical University
Iraq
shamil.ramadhan@ntu.edu.iq

Firas Abdulrahman Yousif
Department of Computer Science, Faculty of Education
University of Al- Hamdaniya
Iraq
firasabdulrahman@uohamdaniya.edu.iq

Firas Shawkat Hamid
Department of Medical Instrumentation Techniques
Northern Technical University
Iraq
dr.firas.hamid@ntu.edu.iq

Article history:
Received 30.11.2025, Accepted 20.12.2025

Abstract

Hand gestures are essential in human communication, and recent technological advancements aim to extend this natural interaction to human-computer interfaces. This paper introduces a real-time volume control system that enables users to adjust sound levels using hand gestures, eliminating the need for physical components. The system relies on a standard computer camera to detect and interpret finger patterns, using MediaPipe to identify hand landmarks and OpenCV in Python to analyze live video frames. By calculating the distance between fingers, the system adjusts the volume through pycaw, offering a full range from minimum to maximum. A key innovation is its ability to address two major challenges: first, compensating for varying hand-camera distances (ranging from 40 to 300 cm), and second, defining a specific pattern of fingers to ensure only intentional gestures affect volume control. This prevents unintended hand movements from interfering with system operation. The gesture recognition process is enhanced through a set of algorithms that refine detection and improve reliability. With an impressive accuracy of 97.3%, the proposed system demonstrates the potential of gesture-based control in real-time applications, offering a seamless and intuitive user experience. It represents a significant step toward more natural and efficient human-machine interaction.

Key words

Gestures, Hand detection, Hand distance, OpenCV, Volume control, Webcam

1 Introduction

Users can communicate with machines using hands, facial expressions, voice, and touch. Gesture recognition has attracted much research and interest in human-computer interaction, pattern recognition, and vision in computers. A research goal is finding a relationship between hand gestures and volume control. Prevent all hand movements from being volume-controlled and determine only one pattern to control. Create a relationship between the location of the hand and the camera. In order to create a robust system that can withstand functioning under the range of environmental parameters such variations in illumination and the existence of background clutter among others, a combined approach should be taken incorporating adaptive sensor calibration, resilient feature extraction and context-aware data augmentation. Also, strict testing procedures that reflects on the real world variability should be used, thus making the system to be dependable under the varied and dynamic conditions. Evaluate and assess the system: Following implementation, testing and evaluation are required to verify its accuracy and dependability. The ground truth can be contrasted to the system's findings, which can be assessed using various hand gestures and volume levels. Shape recognition is one of the leading activities at the machine vision field. It also has a central niche in the wider field of computer science scholarship that includes the field of image understanding and pattern recognition [Khurana, 2023]. Most recently, an interesting development has been made into human gesture recognition through deep learning, with results where innovative models combine Transformer-based designs with 3D pose estimation, hence signalling a significant improvement in realistically deployed settings. This new development greatly grants the work efficiency of interactive systems with visions [Hung, 2024]. Gesture recognition, with pattern recognition, depicts a new and active field of academic research. Such things as hand motions have the potential to add another layer to the functional effectiveness of the non-contact computing systems [Purushotham et al., 2023a]. Gestures are a highly effective form of communication that can substantially improve human-computer interaction (HCI). A hand-tracking mechanism is an essential requirement for every gesture recognition system. Users may control the volume of their devices remotely [Tiwari et al., 2023]. Hand gestures constitute a mode of nonverbal communication applicable in various domains, including interactions among deaf-mute individuals, robotic control, human-computer interaction (HCI), home automation, and medical applications [Oudah et al., 2020]. Hand gesture recognition is used in numerous applications, including HCI, robotics, sign language, digit and alphanumeric value, and home automation [Kedari et al., 2022]. Devices and applications that are not volume controllers, such as home automation systems, video game consoles, and virtual reality displays, can be included in the extension of human-computer interaction. This presents potential opportunities for future research and development in the areas of gesture recognition and the interaction between humans and computers [Tiwari et al., 2023]. Detection systems can be classified into two categories: systems of contactless detection, which operate at a distance from the hand without physical contact, and wearable detection systems, which typically employ many sensors in close range to the hand [Abir et al., 2021]. . People use various methods to express meaning; they may articulate, gesture, or compose to share their thoughts with others. However, the deaf people cannot express themselves using verbal communication. As a result, they are using sign language, the modality that has to be based on the embodied and manual gestures. Hand movement is a fundamental element of the sign language, as far as the particular arrangement of hands dictates a particular semantic message [Narayanpethkar et al., 2023]. Sign language is an imperative linguistic object, which supports substantial and continuous communication between the communities of the deaf or the mute communities and the larger populace. However, there still exists a significant barrier in communication between these communities and the rest of the population. This project will mainly focus on reducing this barrier, and thus improving the experiences of deaf or mute people in the rest of the society [Faisal et al., 2022]. The main barrier the aged and disabled populations have to face in their ever-digitized environment is communication with computing devices. Recently, it has placed hand gesture recognition as one of the brightest examples of the most naturalistic human computer interfaces in modern software engineering [Cicirelli et al., 2021]. It has been empirically indicated that such systems prove to be more effective in those settings that are defined by the presence of auditory turbulence, in which traditional manual controls tend to fail, thus underscoring the utility of the concept of acoustically controlled volume management as an interesting development in the sphere of assistive technologies [Tiwari et al., 2023]. Hand gestures may be either static or dynamic, a static gesture is when a particular hand position is established and a dynamic gesture entails a series of hand motions e.g. waving [Oudah et al., 2020]. A vision-based system was created to understand hand motions by regulating volume with color and motion detection. It used a camera to record and analyze hand movements, then adjusted the volume based on the recognized gesture. However, limitations included the system's ability to recognize only a limited variety of movements [Tiwari et al., 2023]. A different research study suggests using depth-sensing cameras to detect hand movements and gestures to adjust volume. The volume was measured accordingly with the gesture that was detected and machine-learning algorithms were used in the system to identify various hand motions. Despite being an efficient system, it required specialised equipment to operate, hence making the system costly as well as not accessible to the average user [Tiwari et al., 2023]. The exploration of wearable computers to control the volume has been one of the areas of intense focus of a plethora of academic research questions in the field of human-computer interaction. In a significant investigation, scientists suggested the introduction of wristwatch that could be used as a close interface in adjusting the level of audio equipment. Through combining verbal instructions and tactile gestures in a intuitively developed system, users could attain a smooth and intuitive system of volume control [Tiwari et al., 2023]. The data glove sensor arrived and with-it people could begin to steer computers - moving their hands. It supplied basic commands for computer interfaces by using a variety of sensors to detect hand motion and position. The data was processed on a computer connected to the glove, providing a potentially portable device with a sensor and a microcontroller [Purushotham et al., 2023b]. Introduced a system for recognizing American Sign Language (ASL) movements using wearable sensors and deep learning algorithms. This system uses wearable sensors to capture hand and body movements, and deep learning algorithms process the collected data [V and Sanskruti, 2023]. The accuracy and speed of picture classification within an operational environment are vital for the successful completion of tasks in intelligent control systems. Similar to mobile robot systems that operate in uncertain situations induced by measurement inaccuracies [Kuchmin, 2025]. Proposed real-time human pose classification using depth sensor skeletal data. However, calibrating a binocular camera is difficult, and Leap Motion only recognizes distances ranging from 2.5 to 60 cm. No celebration or long-distance recognition [Choubik and Mahmoudi, 2016]. The treatments outlined have limitations that make them unsuitable for elderly persons due to discomfort, confusion, and the possibility of skin damage. Gloves might be difficult for older persons with chronic diseases, and some sensors are pricey [Oudah et al., 2020]. Gestures face three main challenges: hand segmentation, hand form feature representation, and gesture sequence detection [Aly and Aly, 2020]. Classification makes use of the pixel-wise depth difference between the recorded templates and the observed gesture. Due of the limited resolution of the ToF camera, this method requires the user to be within 1 meter of it [Kapuscinski et al., 2015]. The problematic aspect of these systems is the collection of backdrop photographs or videos during the input process, namely the user's hand motions. Furthermore, lighting conditions might negatively impact the quality of the captured input, complicating gesture detection. The procedure for identifying a connected region inside an image based on attributes such as color, intensity, and pixel relationships, sometimes referred to as patterns, is known as segmentation [Saboo and Singha, 2021]. Research reveals that the detection range of the depth camera is inadequate for gesture identification when the distance between the Kinect sensor and the human hand surpasses 2.5 meters [Ma and Peng, 2018]. Ahmed et al. Employed a chip with integrated radars for the identification and classification of hand movements [Ahmed et al., 2021]. This chip is ineffective for gesture detection at a distance of one meter [Faisal et al., 2022]. The initial stage in any hand-processing system is detecting and finding the hand in real-time footage from the camera. Detecting a hand might be challenging because of its position, orientation, placement, and scale variation [Dixit, 2023]. Object detection and tracking is one of the most complex phases in any image processing application. The challenges that arise during hand gesture recognition are hand form variation, presence of many persons, light variation or occlusion with background [Saboo and Singha, 2021]. The suggested approach uses skin color segmentation and blob analysis to locate the hand in the video frame. A machine learning model trained on a dataset of hand gestures recognizes the hand gestures [Rajput and Kumar, 2024]. Different light levels in the room can also cause changes. Gesture recognition often requires multiple processing layers, including image acquisition, preprocessing, feature extraction, and gesture recognition [Rajput and Kumar, 2024]. Creating a gesture recognition system is complex and involves two significant challenges. The first is discovering the human hand. The second is using the webcam to capture the user's hands in real-time video [Tamilkodi et al., 2024]. The camera must record video frame by frame. As part of image processing, captured images are filtered and edited. Feature extraction is a technique that extracts features (such as hand contours) from hand images; gesture recognition is a technology that extracts features and recognizes actions [Aly and Aly, 2020]. This study aims to find a relationship between the distance between the thumb and index fingers in the hand and the remote sound control. An evaluation of this model is needed to execute it in real-time. And Assign a specific pattern to achieve that. And suggested an algorithm to solve the distance issue.

2 Methodology

The system can record and examine the user's hand movements in real time through OpenCV's use for camera input and image processing. While PyCaw offers a Python interface to the Windows CoreAudio API, enabling the system to communicate with the computer's audio system, NumPy is utilized for data processing and manipulation. In the beginning, it is necessary to know the algorithm for the hand, the arrangement of the fingers, and how it works. Identified Mediapipe to detect the palm and identify 21 hand landmarks based on the action. Fig. 1 shows the hand landmark chart.

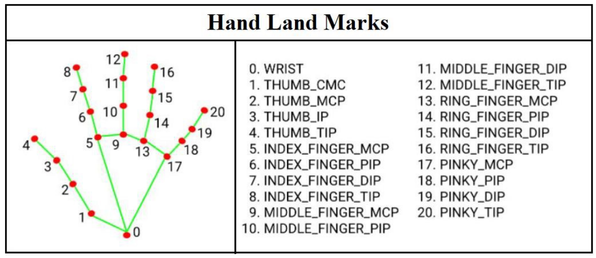

Figure 1. Hand land marks chart

The proposed system implemented a gesture-based volume control system in Python, combining real-time hand tracking, distance estimation, and audio control. Below is a detailed description of the process for every challenge:

2.1 Implementing and Assigning One Pattern for Volume Control

Implement pattern recognition for audio control. This approach involves creating a set pattern of hand gestures that the system can recognize and interpret. Pattern recognition allows the system to identify specific gestures. By assigning a specific pattern of hand movements to correspond with volume control, users would have a consistent and intuitive method for adjusting the volume through the following steps of a suggested algorithm: Image acquisition and hand detection through live video streaming captured using a standard webcam. The CVZone. HandTrackingModule detected a single hand per frame with a minimum confidence of 0.5. The frames were flipped horizontally for mirror-image interaction, converted to RGB, and processed by MediaPipe Hands to extract 21 landmark coordinates for fingertip localization in Fig. 2. One frame of the hand shows how it appears before the special gesture is activated.

Assign only one pattern for volume control in real-time. The pattern is determined by raising two fingers, the thumb tip (landmark 4) and index-fingertip (landmark 8) and the pinky-fingertip (landmark 20) with both

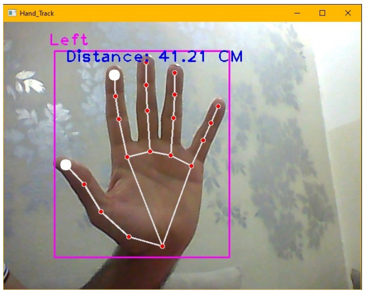

Figure 2. One frame of the hand

the middle (landmark 12) and ring (landmark 16) fingers are folded. This process prevents sound control from being affected by any other movements of the hand. Therefore, the volume will only be controlled if this pattern is applied exclusively. Determined the thumb and forefinger's locations for the hand. And created a line that connects them and calculates its center as the distance equation below. This line will later represent a relation with volume range.

$$\text{Dis} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{1}$$

Fig. 3. Shown is the hand form after the special gesture is activated.

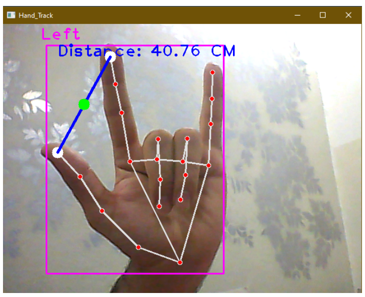

Figure 3. Assign a pattern

2.2 Solving the Distance Challenge in Gesture Recognition

A primary problem in gesture recognition is accurately identifying and interpreting hand gestures at different distances from the camera. The proposed System could solve the hand distance issue from the camera, as the algorithm could be trained to recognize gestures from varying distances by detecting and calculating the hand's distance from the camera and suggesting an equation to overcome this challenge. And Steps of a suggested algorithm are:

Will be using a set of below equations to provide additional information about the hand's distance from the camera, allowing us to normalize and calibrate the gestures for different distances. Based on this distance information, the system will adjust the recognition and interpretation of hand gestures accordingly. This will guarantee that users may precisely regulate the loudness, regardless of their distance from the camera. To calculate the focal length for later distance estimation, we used a previously captured image of a hand 9 cm width (Width_Known) placed 50 cm (Distance_Known) away as a reference. The focal length was then computed as follows:

$$\text{Focal\_Length}_{\text{px}} = \frac{\text{Width}_{\text{ref}} \times \text{Distance}_{\text{known}}}{\text{Width}_{\text{known}}} \tag{2}$$

The code reads a pre-calculated ratio (width in the reference image = 4800px) from ref-config.txt, which was previously extracted from the hand image as a reference to ensure consistent calibration across sessions.

Raw Distance Estimation in each frame, the bounding-box width of the detected hand (Width_Current) was measured in pixels by the MediaPipe Hands library. Then the raw estimate distance was computed by inverting the proportional relationship:

$$\text{Distance}_{\text{Estimate-px}} = \frac{\text{Width}_{\text{known}} \times \text{Focal\_Length}_{\text{px}}}{\text{Width}_{\text{current}}} \tag{3}$$

A set of distance points will be selected to represent the actual distance between the hand and camera at (50-100-150-200-250-300) cm, allowing for a comparison with the estimated distances and preparation of their data for correction and calibration in the next linear correction model step. To mitigate systematic non-linear errors - especially in extended ranges - a linear correction model with empirically determined coefficients (a = 1.15056, b = -0.84349) was applied:

$$\text{Distance}_{\text{Corrected}} = a \times \text{Distance}_{\text{Estimate}} + b \tag{4}$$

Gesture Normalization and Volume Mapping by converting the pixel distance to centimeters by accounting for the corrected hand-to-camera distance and the pixel-per-centimeter ratio

$$\text{Pixel Per Cm} = \frac{\text{Width}_{\text{ref}}}{\text{width}_{\text{Known}}} \tag{5}$$

To address the approximation error associated with the linear correction model in Equ. (5), we conducted a robustness analysis under varying noise levels. Following the methodology discussed in recent literature, the error bounds were evaluated by simulating measurement uncertainties from 0% to 100% of the signal level. This ensures that the approximation error remains systematically overbounded within a confidence interval suitable for real-time interaction. While finding an optimal solution is theoretically possible through 3D pose estimation and transformer-based fusion modules, our empirical linear model provides the necessary computational efficiency for mobile and real-time hand gesture volume control.

Then, the physical length is then normalized to [0,1] based on experimentally determined bounds (1.5 cm-8 cm). The normalized value is scaled to a percentage (0%–100%) and linearly interpolated to the system's audio dB range extracted via pycaw.GetVolumeRange() as roughly -65.25dB up to 0dB. The resulting volume level is set in real time using volume.SetMasterVolumeLevel().

The corrected distance will be detected and shown as the value on the screen's background that represents how far a hand is from the camera. Fig. 4 Flowchart for the steps to achieve it.

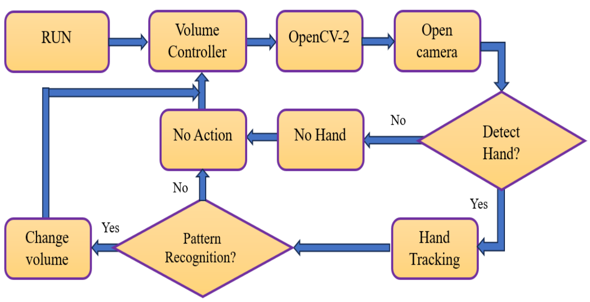

Figure 4. Block diagram for proceeding steps

3 Results

The objective is to adjust the volume of a computer device without the need for physical interaction using a mouse or keyboard. OpenCV is widely used for working with images and for computer vision tasks. Every frame gives the user an immediate picture of what is happening:

• The corrected distance (hand-to-camera) in (cm).
• Current volume percentage.

Graphical overlays, which include lines, circles, and text-based annotations of whatever is being scanned are color-coded to distinguish between legitimate gestures and, by extension, out out of range status hence providing a quicker feedback as to how walletable is the system control regime or as to the faithfulness of the gestures. The system incorporates seamlessly computer-vision calibration, empirical error fixation, in addition to gesture-normalized audio control, hence supporting a reliable model to hands free volume regulation in the field of application. The system function does not initially activate without moving fingers according to the pattern, as point No. 2 in the methodology (Implementing and Assigning One Pattern for Volume Control). Fig. 5 shows the pattern assigned.

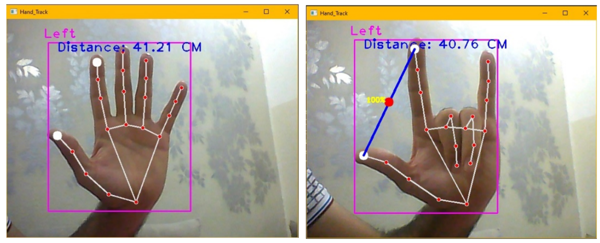

Figure 5. System operation activation

The second challenge is the distance between the hand and the camera. The system was tested at various distances to ensure its ability to solve this challenge.

The focal length calibration was computed using a reference hand image at 50 cm with a known palm width of 9.0 cm. The derived focal length exhibited consistent depth estimation across multiple sessions. Fig. 6, Fig. 7, and Fig. 8 below show the system performance tested at two different distances (about 40 cm and 300 cm) to compare the system's ability to detect actual volume values at (0, 50, and 100 percent) for volume.

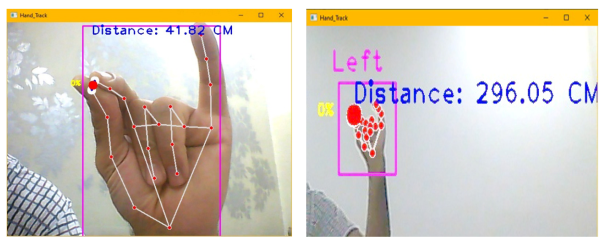

Figure 6. System tested across 40–300 cm at 0% of total volume

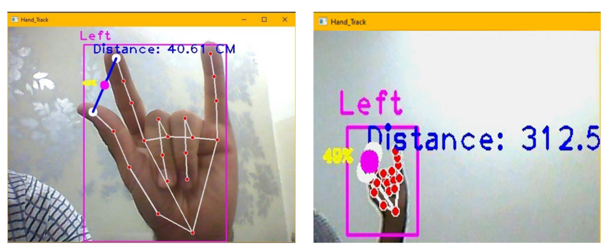

Figure 7. System tested over 40–300 cm range at 49% of total volume

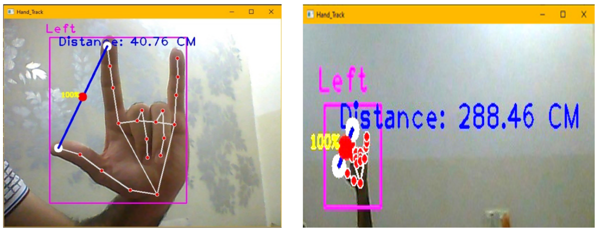

Figure 8. System tested across 40–300 cm range at full volume

During every test we ran, we checked the results against the actual hand position at multiple distances. This showed how well the system works out the right volume on the spot. With the implementation of a linear correction model, the distance estimates became much more accurate. Table 1 shows how well the calibration worked.

Table 1. Comparison of Actual, Estimated, and Corrected Distances

$\text{Distance}_{\text{Actual}}$	$\text{Distance}_{\text{Estimated}}$	$\text{Distance}_{\text{Corrected}}$
50	48	54.38
100	87	99.26
150	133	152.29
200	165	189.97
250	218	250.12
300	266	305.12

Where:
Actual Distance: The real distance from the hand to the camera. Estimated Distance: The distance the vision system computes before it is adjusted. Corrected Distance: The distance value that results once calibration or correction.

Before the correction, the measured distances were consistently wrong when compared with the real values - the error grew with range and became obvious beyond 250 cm revealing that the raw model bends the truth in a nonlinear way. Figure 9 shows the real distance, the raw estimate plus the corrected value side by side.

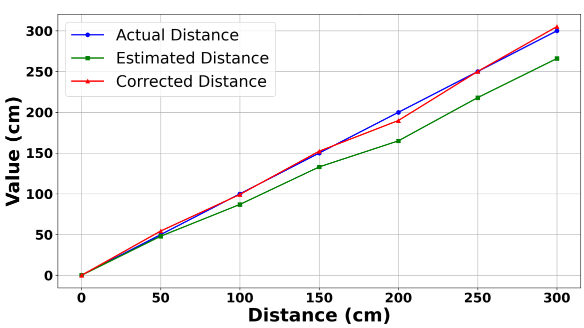

Figure 9. Comparison of actual, estimated, and corrected distances

Post-correction, the Mean Absolute Error (MAE) was reduced to 4.03 cm, and the Root Mean Square Error (RMSE) dropped to 5.03 cm, reflecting a significant reduction in variance between predicted and actual values. The coefficient of determination (R^2) reached 0.996, confirming a strong linear relationship between corrected and true distances. These results demonstrate the model's ability to compensate for distance-dependent measurement errors and enhance the reliability of the system under real-world conditions. The overall system accuracy reached 97.3%, validating the effectiveness of the correction strategy across the full operational range. Table 2 shows the statistical evaluation metrics for the proposed system.

Table 2. Statistical Evaluation Metrics

Metric	**Value**
Accuracy	97.3%
RMSE	5.03 cm
MAE	4.03 cm
Standard Deviation	4.91 cm
R^2	0.996

The new audio interface lets a person raise or lower the sound - moving two fingertips apart or together. A webcam watched the hand under steady light, read the joint positions and worked out how far the fingers were from each other so the volume changed at once plus a small on screen bar moved with it.

The experiments show that the present 2D system maps volume in real time and still recognises gestures reliably, even when faced with uncertain data arising from variable camera distances. This stability under informational uncertainty makes the proposed framework particularly suitable for practical, real-world applications where user positioning is not fixed. However, following trends in recent literature 2, integrating 3D Pose Estimation could further enhance future accuracy. Such advanced spatial modeling would enable more sophisticated noise removal in video sequences than traditional 2D methods, potentially offering finer-grained control in complex environments.

Table 3 is a comparative overview of modern research in the area of gesture-controlled audio interface. We have made an important step forward in our current work, insofar as the accuracy for a distance interval of

40-300cm is 97.3% when using a single finger-pattern scheme. This technique is a skillful way to overcome both distance and volume modulation issues by using different gesture patterns. Ma and Peng (2018), on the other hand, achieved a 96% suspend accuracy at a smaller range of 50-200 cm, which was based on the position of fingertips, mainly to control volume. Most recent studies (2023), such as those of Tiwari et al., Rajput et al., Dixit et al., and Khurana et al., used thumb and index-finger gestures with different precision and operational ranges being reported. Khurana and colleagues reached 99% accuracy - yet their setup was built only for a virtual mouse that adjusted loudness and screen brightness. The study presented here differs because it widens the range of tasks plus supplies a gesture recognition system aimed at audio commands and able to adjust to new needs.

Table 3. Comparison of gesture-based volume control

Study	Acc.	Range cm	Gesture
This Study	97.3%	40–300	Custom
Ma & Peng	96%	50–200	Fingertip
Tiwari et al.	95%	–	Thumb+Index
Rajput et al.	–	–	Thumb+Index
Dixit	–	–	Thumb+Index
Khurana	99%	–	Thumb+Index

Tests show the new system is more accurate and works over a larger space than earlier audio systems that were driven by gestures. With an accuracy rate of 97.3% and, it responds when the user stands between 40 cm to 300 cm away - it performs well in rooms of many sizes. Compared to previous studies, which were mostly focused on volume modulation through crude gestures such as thumb-index combinations, the employed system uses a custom finger pattern to control both distance and volume harmoniously.

4 CONCLUSIONS

Hand gestures have become a growing trend in nonverbal communication across a wide range of fields, including human-computer interaction (HCI) and home-automation systems. The utility of hand gestures for controlling devices is underscored by the proposed system, which combines computer vision and gesture-recognition algorithms, enabling users to adjust the volume with a specific gesture. The system will automatically identify and analyze gestures by volume and convert them into a set of volume-control instructions. Users can raise or lower the volume using gesture controls. The innovation of this proposed system is that, to the authors' knowledge, most of the literature examined offers no such advanced experimental design. Although gloves can provide accurate measurements of hand shape, they are too bulky to carry and must be wired. A webcam is an interface that enables a new form of human-computer communication in which no physical contact occurs. Moreover, it proposes an algorithm that can correctly determine the distance between the user's hand and the camera. The calibration procedure suggested provides a strictly hypothetically tested roadmap to high-fidelity distance estimation and, as such, is consistent with the broader process of creating robust computer-vision systems that are highly accurate and reproduce such outcomes. Despite these developments, most of the existing literature lacks a comprehensive discussion of distance estimation. However, the described technique has significant potential in the large-scale deployment in the domestic ecosystem of devices in the near future. Moreover, the technology provides new opportunities for people with disabilities, enhances the overall user experience, and increases productivity.

References

Khurana, S. (2023). Design and implementation of virtual mouse with volume/brightness control using hand gestures. *Soft Computing Research Society eBooks*. https://doi.org/10.56155/978-81-955020-3-5-06

Hung, L. V., Khoa, T. D. M., Han, L. H. N., Son, T. V. H. (2024). A deep learning framework for gym-gesture recognition using the combination of transformer and 3D pose estimation. *Cybernetics and Physics*, 13(2):161–167. https://doi.org/10.35470/2226-4116-2024-13-2-161-167

Purushotham, P., Bennet, B. S., Vijayakarthik, P., Kiran, A., Pravallika, M., Shivani, Y., et al. (2023). PC automation using hand gestures. In *Proc. Int. Conf. Computer Communication and Informatics (ICCCI)*, pp. 1–4. https://doi.org/10.1109/ICCCI56745.2023.10128210

Tiwari, S., Mishra, A., Kukreja, D., Yadav, A. L. (2023). Volume controller using hand gestures. In *Proc. Int. Conf. Computing Communication and Networking Technologies (ICCCNT)*, pp. 1–6. https://doi.org/10.1109/ICCCNT56998.2023.10308134

Oudah, M., Al-Naji, A., Chahl, J. (2020). Hand gesture recognition based on computer vision: A review of techniques. *Journal of Imaging*, 6(8), 73. https://doi.org/10.3390/JIMAGING6080073

Kedari, P., Kadam, S., Prasad, R. (2022). Controlling computer using hand gestures. *Int. Res. J. Eng. Technol. (IRJET)*, 5(3), 1050–1054. https://doi.org/10.46253/j.mr.v5i3.a2

Abir, F. F., Faisal, M. A. A., Shahid, O., Ahmed, M. U. (2021). Contactless human activity analysis: An overview of different modalities. *Intelligent Systems Reference Library*, 200, 83–112. https://doi.org/10.1007/978-3-030-68590-4$_3$

Narayanpethkar, S., Makude, A., et al. (2023). Computer vision based media control using hand gestures. *Int. J. Res. Appl. Sci. Eng. Technol.*, 11(v), 6642–6646. https://doi.org/10.22214/ijraset.2023.52881

Faisal, M. A. A., Abir, F. F., Ahmed, M. U., Ahad, M. A. R. (2022). Exploiting domain transformation and deep learning for hand gesture recognition using a low-cost dataglove. *Scientific Reports*, 12(1), 1–15. https://doi.org/10.1038/s41598-022-25108-2

Cicirelli, G., Marani, R., Petitti, A., Milella, A., D'Orazio, T. (2021). Ambient assisted living: A review of technologies, methodologies and future perspectives for healthy aging of population. *Sensors*, 21(10), 3549. https://doi.org/10.3390/s21103549

Purushotham, P., Madhu, R., Vinod, T., Shekar, K. (2023). PC automation using hand gestures using machine learning. *Int. J. Adv. Res. Sci. Commun. Technol.*, 3(1), 92–97. https://doi.org/10.48175/ijarsct-12918

V, A. E., Sanskruti, S. (2023). Hand gesture volume control using machine learning. 5(4), 1894–1896. https://doi.org/10.35629/5252-050418941896

Kuchmin, A. Y., Tarasova, I. L. (2025). Image formation in a mobile robot's operating environment and classification using neural networks and logical-linguistic algorithms. *Cybernetics and Physics*, 14(3):248–254. https://doi.org/10.35470/2226-4116-2025-14-3-248-254

Choubik, Y., Mahmoudi, A. (2016). Machine learning for real time poses classification using Kinect skeleton data. In *Proc. Int. Conf. Computer Graphics, Imaging and Visualization (CGiV)*, pp. 307–311. https://doi.org/10.1109/CGiV.2016.66

Aly, S., Aly, W. (2020). DeepArSLR: A novel signer-independent deep learning framework for isolated Arabic sign language gestures recognition. *IEEE Access*, 8, 83199–83212. https://doi.org/10.1109/ACCESS.2020.2990699

Kapuscinski, T., Oszust, M., Wysocki, M., Warchol, D. (2015). Recognition of hand gestures observed by depth cameras. *Int. J. Adv. Robotic Syst.*, 12(4), 36. https://doi.org/10.5772/60091

Saboo, S., Singha, J. (2021). Vision-based two-level hand tracking system for dynamic hand gestures in indoor environment. *Multimed. Tools Appl.*, 80(13), 20579–20598. https://doi.org/10.1007/s11042-021-10669-7

Ma, X., Peng, J. (2018). Kinect sensor-based long-distance hand gesture recognition and fingertip detection with depth information. *Journal of Sensors*, 2018(1), 5809769. https://doi.org/10.1155/2018/5809769

Ahmed, S., Kallu, K. D., Ahmed, S., Cho, S. H. (2021). Hand gestures recognition using radar sensors for human-computer interaction: A review. *Remote Sensing*, 13, 1–24. https://doi.org/10.3390/rs13030527

Dixit, Dr. (2023). Hand gesture recognition and cursor control. *Int. J. Res. Appl. Sci. Eng. Technol.*, 11(6), 625–629. https://doi.org/10.22214/ijraset.2023.53665

Rajput, A., Kumar, A. (2024). Volume control using gesture recognition system. 24(6), 19. https://doi.org/10.22937/IJCSNS.2024.24.6.19

Tamilkodi, R., Madhuri, N., Dhanushkumar, G., Dileepkumar, G., Rajkumar, G., Sandeep, Y. (2024). Hand gesture recognition and volume control. In *Proc. Int. Conf. Atlantis Press International BV*, pp. 1002–1012. http://dx.doi.org/10.2991/978-94-6463-471-6$_9$6

Rajput, A., Sharma, U., Kumar, A. (2023). Volume controlled by hand gesture recognition. *ECB*, 12(SI5A), 6581–6587. https://doi.org/10.48047/ecb/2023.12.si5a.0591