IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 11, No. 2, June 2022, pp. 709~716
ISSN: 2252-8938, DOI: 10.11591/ijai.v11.i2.pp709-716

Error detection and comparison of gesture control technologies

Aditya Prasad Mahapatra[1], Bishweashwar Sukla[1], Harikrishnan KM[1], Debani Prasad Mishra[2], Surender Reddy Salkuti[3]

[1]Department of Electrical and Electronics Engineering, International Institute of Information Technology (IIIT), Bhubaneswar, Odisha, India
[2]Department of Electrical Engineering, International Institute of Information Technology (IIIT), Bhubaneswar, Odisha, India
[3]Department of Railroad and Electrical Engineering, Woosong University, Daejeon, Republic of Korea

Article Info

Article history:

Received Oct 27, 2021
Revised Dec 24, 2021
Accepted Jan 11, 2022

Keywords:

Arduino
Errors
Gesture control
IR sensors
Ultrasonic sensors

ABSTRACT

This paper showcases the study and observation on the error occurrence in gesture control technologies. An arduino-based gesture control environment has been developed by using the arduino board to use motion gestures to control the contents on the screen. This environment is made using position sensitive diodes as sensing devices, arduino as a micro-controller and Python to execute commands in the system. It is performed on 2 different software applications namely Google Chrome and VideoLAN Client Media Player. In Google Chrome gestures are used to traverse between tabs and also move up and down within a web page, whereas in VideoLAN Client Media Player gestures are used to control the volume and speed. Through this, the difference between two technologies i.e., infrared and ultrasonic are worked and compared. Various data visualization cues are prepared to better understand the error and the factors causing it. Thorough investigation of factors affecting the error has been done using our observation. The future of this technology and its limitations have been also discussed.

Corresponding Author:

Surender Reddy Salkuti
Department of Railroad and Electrical Engineering, Woosong University
17-2, Jayang-Dong, Dong-Gu, Daejeon-34606, Republic of Korea
Email: surender@wsu.ac.kr

1. INTRODUCTION

This paper illustrates error occurrences and drawbacks of using ultrasonic technology for gesture control through the project Gesture control for computers using ultrasonic sensors [1]. Earlier interaction between computer and human was only limited to keyboard and mouse, but today you can interact with gestures i.e facial recognition, and hand gestures [2]. Gestures and image recognition are used in human skin detection [3]. Gestures being the primitive form of communication within humans, these are highly useful for computer interaction [4], [5]. Dynamic gesture recognition of human gestures makes up an important means of user-machine interface [6]. Gesture controls are used in computer vision for human computer interaction with electromyography [7]–[9]. Automated recognition of human gesture is a boon for user interfacing [10]. Human hand gestures have become more natural and with the help of wireless communication [11], various gestures of the computer can be controlled using these sensors and hand recognition [12]. Gestures are detected using optical and motion technology by the means of bluetooth, acoustics, tactiles, or infrared waves [13]. In the vision based system, motion and skin color information is utilized to detect waving hands requesting control commands [14]. Research is still very active in unresolved challenges such as reliable input such as identification of gestures, sensitivity to variations in size, shape, speed and issues due to occlusion [15]. Making a cost efficient highly accurate model for a gesture control system is essential for the future [16].

Vision-based gesture recognition models are based on gesture detection and classification, feature extraction and pre-programmed template execution process [17]. Various gestures of the computer can be controlled using these sensors and hand recognition. Hand recognition based on visual technology is an active area of research from ages [18]–[20] These approaches are based on distance and accuracy which are quite flexible, but sometimes they make wrong detection on lighting changes and reflection. There are not many practical hand gesture recognition systems that use visual technologies due its inaccuracy despite its cost efficiency [21]. One of the prime features of using hand gestures is to interact with the computer as an input unit [22], [23]. So, this paper presents the study on the most cost-efficient models and ways to increase their accuracy. These models are precisely demonstrated using two techniques namely infrared and ultrasonic sensors.

In the present work, each model was provided with six numbers of gestures to manipulate the screen motions and gestures were set and programmed accordingly in the Arduino integrated development environment and using python programming language, commands were given to the display [24]. Distance between sensor and hand are measured using sensors, which helps to detect the type of movement and then the computer shows the result [25]–[27]. Then accuracy of gestures was quantified and represented in pie charts using proper data visualization techniques in Jupyter notebook. Differences in accuracy between two technologies were quantified and results were concluded. Figure 1 shows the flow of function in an ideal gesture control system. Input is given to the system which is then read by the sensing devices. Then the microcontroller interprets and processes the information. After interpretation a command or a signal is given to the system. This command or signal interacts with the output or display such that favorable changes are made in the display or output. Figure 2 shows the flow of function in a proximity based ideal gesture control system. Gesture is given as input to the system which is then read by the proximity sensors. Then the microcontroller interprets and processes the information. After interpretation a command or a signal is given to the system. This command or signal interacts with the output or display such that favorable changes are made in the output.

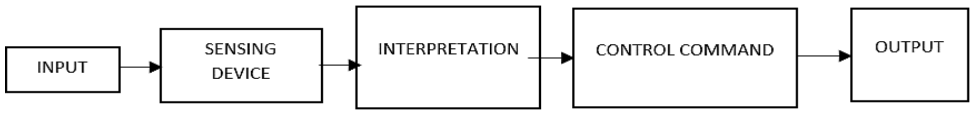

Figure 1. Flow chart for an ideal gesture control device

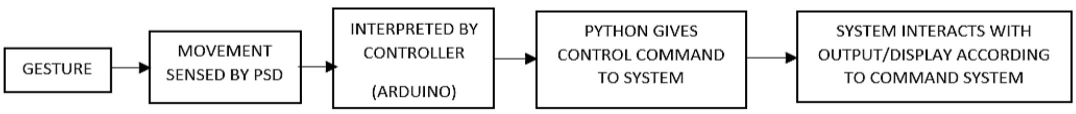

Figure 2. Flow chart for proximity sensor-based gesture control system

2. RESEARCH METHOD

2.1. Gesture control system using ultrasonic sensor

Different components are required to develop the gesture control system using ultrasonic sensors. Those are Arduino UNOx1, ultrasonic sensorsx2, USB cable (for Arduino), a device for programming and display (PC) and jump wires. Figures 3 and 4 depicts the circuit diagram for ultrasonic sensor-based gesture control systems.

The trigger and echo pins of the first ultrasonic sensor are connected with pins 8 and 9 of the Arduino while pins 2 and 3 of the Arduino are connected with the trigger and echo pins of the second ultrasonic sensor. The two ultrasonic sensors are placed on two corners of the display, i.e., the left and right side of the PC display in our case. If they are placed too close then the signal will interfere and wrong results will come. After checking the connections, serial inputs are set in the Arduino IDE representing each input point in the circuit. In the loop function separate functions for the left, right, up and down gestures are defined. The distance is calculated using the principle of reflection of sound. When a sound wave travels back to us after reflecting a surface, the distance is given by,

$$Distance = Speed \times (time/2) \quad (1)$$

Half the total time is taken because sound covers the same distance to travel twice. Now, the speed of sound at sea level is 343 m/s.

$$Distance = (343/2) \times time \quad (in\ m) \tag{2}$$

Therefore,

$$Distance = (17150) \times time \quad (in\ cm) \tag{3}$$

In the next step a particular range is defined at which the hand can move towards and away from the sensor. Those distances are taken to be 10 to 30 cm to avoid any other noise or disturbance that could occur in the background while using this technology. Now since the boundary at which our system works is decided, now the system is set to form our command for the right- and left-hand movements. Commands are set in such a way that sensors identify the movement of hands from one sensor to another by using reflected sound waves on both the sensors in a given time interval.

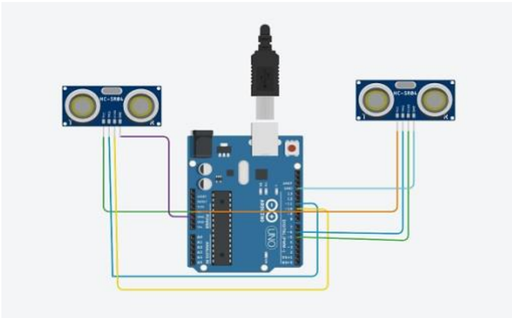

Figure 3. Circuit diagram for ultrasonic sensor-based gesture control system

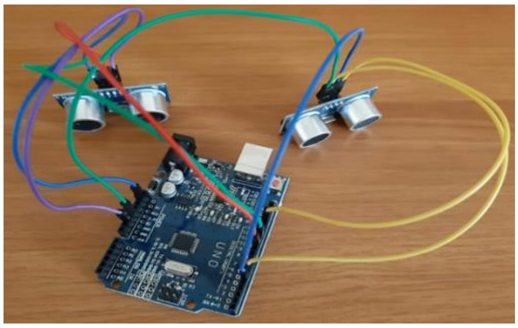

Figure 4. Circuit for ultrasonic sensor-based gesture control systems

2.2. Gesture control system using infrared sensor

Components required to develop the gesture control system using infrared sensors [28] are: Arduino UNOx1, 2xIR sensor, USB cable (for Arduino), a device for programming and display (PC) and jump wires. Figures 5 and 6 depict the circuit diagram for an IR based gesture control system. passive infrared sensors are used for enhancing sensing reliability [29].

Associated a jumper wire in the collector pin of the integrated circuit to utilize this sensor as a simple sensor. To perceive signal developments utilizing IR sensors, a trigger system is utilized. Let two IR sensors be placed on opposite sides of the display and named left-IR and right-IR. Additionally, place your IR sensor such that there is around 3 cm of hole between them. Code has been set up on the arduino according to the principle of reflection of light. 2 sets of infrared sensors each having a receiver and a producing part are prepared. The pins are carefully connected to the arduino circuit board, later these pins are set to arduino IDE. Now functions for different gesture motions are declared. The gesture 1 and gesture 2 are programmed in such a way that the sensor recognizes how far can our arms from and away from itself, the data from the serial monitor is obtained and likewise it is fed to the python program. Meanwhile the programs of gesture 5, gesture 6 and gesture 4 are actuated in such a way that both the sensor checks the vicinity of the object to one another, likewise, these sensors now capture the wave movement of the hand.

2.3. Programming Arduino to detect gestures

Different cases are made to distinguish gestures from each other based on the distance measured by the sensors over a time period which is unique to a particular gesture. Creating different cases for different gestures makes it possible to execute different commands on sensing different gestures which is done by the help of python. Following are the cases:

- Gesture 1: Place your palm in front of the right sensor and move towards yourself (distance of palm from sensors should be between 15 to 35 cm). This motion will scroll down your page in the browser or decrease the volume in the media player.

- Gesture 2: Place your palm in front of the right sensor and move towards sensors (distance of palm from sensors should be between 15 to 35 cm). This motion will scroll up your page in the browser or increase the volume in the media player.
- Gesture 3: Swipe your hand in front of the right ultrasonic sensor. This motion will change your current tab to the Next Tab or fast forward the video running in the media player by 10 seconds.
- Gesture 4: Swipe your hand in front of the left ultrasonic sensor. This motion will change your current Tab to the previous tab or backward the video running in the media player by 10 seconds.
- Gesture 5: Swipe your both palms in front of both sensors simultaneously (left palm a little bit fast).

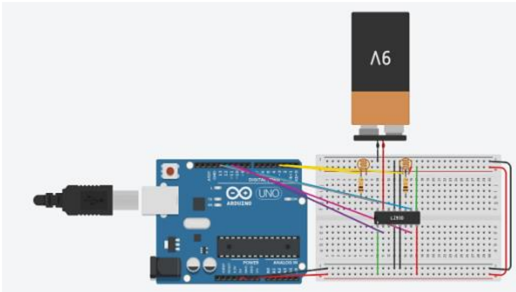

Figure 5. Circuit diagram for IR sensor based gesture control system

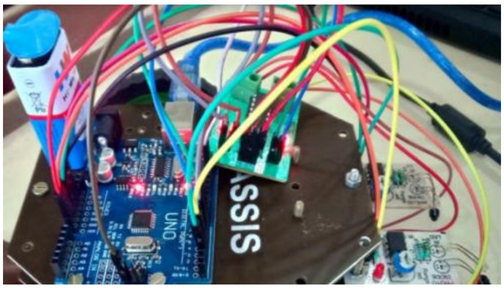

Figure 6. Circuit for IR sensor based gesture control system

This motion switches between tasks (i.e., browser and media player). The automation code for the software actuation is programmed in python, different codes are used for different software which are internet browser and video player respectively. In python code PyAutoGUI is used to read the commands of the arduino IDE provided to the Python code [30].
- if ‘next’ is given as input the system produces pagedown activity.
- if ‘previous’ is given as input the system produces pageup activity.
- if ‘down’ is given as input the system produces scroll down activity.
- if ‘up’ is given as input the system produces scroll up activity.
- if ‘change’ is given as input the system produces change tab activity in the browser.

Control gestures used for the video player are the same as browsers but have different executions. The same PyAutoGUI is used for controlling the output from arduino IDE as well as managing the hotkey functions in the keyboard. The gestures are given below:
- if ‘next’ is given as input the system produces a move forward signal for 10 seconds activity.
- if ‘previous’ is given as input the system produces a move backwards signal for 10 seconds activity.
- if ‘down’ is given as input the system produces a decrease in volume activity.
- if ‘up’ is given as input the system produces an increase in volume activity.
- if ‘change’ is given as input the system produces a pause activity in the video player.

Figure 6 depicts the arduino IDE serial monitor. In Figure 7, this window acts as an interface between the device and the software to exchange information and the message flow to one another through a USB pin. The meaning of the statements in the interface are:
- b‘1\r\n’ indicates true value or positive value
- b‘0\r\n’ indicates false value or negative value

Suppose, for calculating the probability of error for Gesture 1; the output related to Gesture 1 is considered as ‘1’ and other undesired outputs that are related to other gestures are considered as ‘0’. When the program is executed it automatically detects ‘1’ and ‘0’, from there a dataset of these values is formed. Some analytics is performed to find probability and to make the pie charts. This process is done again for all the gestures.

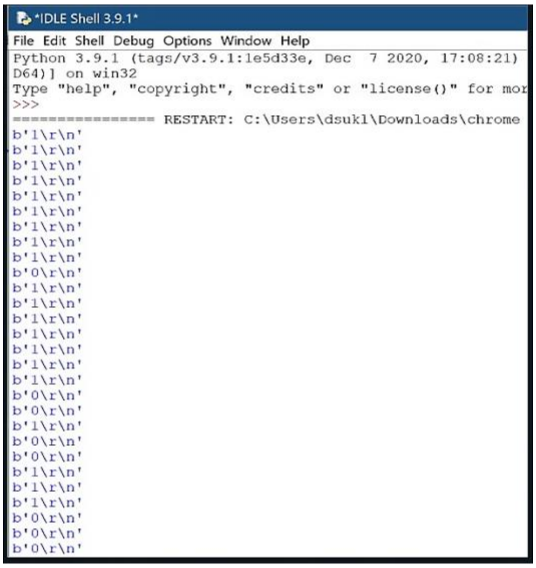

Figure 7. Arduino IDE serial monitor

3. RESULTS AND DISCUSSION

Pie charts showcasing accuracy and faultiness of the models against various gestures were made using data visualization on the observations that were inferred from our models. The gesture control models were experimented on various gestures explained above so that data sets of accurate and faulty commands were obtained using an auto checking program made using serial monitor of arduino IDE. Using this data set, data visualization was done on it using the jupyter notebook.

3.1. Ultrasonic sensor

Experiments performed on ultrasonic sensors showcase the following results. From the observations, it is concluded that using ultrasonic sensors for right swipe gesture 66.3% accuracy and 33.7% faulty observation as shown in Figure 8; for left swipe gesture 85.9% accuracy and 14.1% faulty observation as shown in Figure 9; for scroll down gesture 80.4% accuracy and 19.6% faulty observation as shown in Figure 10; for scroll up gesture 72.9% accuracy and 27.1% faulty observation as shown in Figure 11. The right swipe gesture as shown in Figure 8 was optimized and it’s accuracy was increased by using a barrier between both the sensors as a result, an accuracy of 83.9% was obtained with a 16.1% faulty observation, which is described in Figure 12.

3.2. Infrared sensors

Experiments performed on infrared sensors showcase the following results. From the observations, it is concluded that using infrared sensors for left swipe gesture, 96% accuracy and 4% faulty observation as shown in Figure 13; for right swipe gesture, 93.5% accuracy and 6.5% faulty observation as shown in Figure 14; for scroll up gesture, 95.5% accuracy and 4.5% faulty observation as shown in Figure 15; for scroll down gesture, 94.5% accuracy and 5.5% faulty observation as shown in Figure 16 were obtained. The accuracy percentage of two gesture control models namely, ultrasonic sensor based and infrared sensor-based system for various gestures were observed. Results also show that putting a barrier in between the sensors increases the accuracy of ultrasonic sensors.

As per above results, the following observations were inferred. Ultrasonic sensors were used initially and their observations were analyzed, from which an average accuracy was calculated to be 76.37%. Next, the observations were made upon infrared sensors and average accuracy was calculated as 94.87%. Also, barrier addition resulted in increasing the accuracy of right swipe observations in ultrasonic sensors to about 17.6% of the original value.

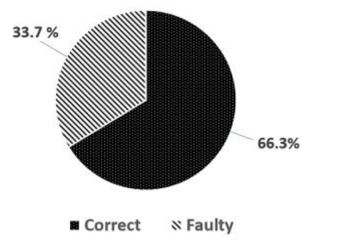

Figure 8. Right swipe observations (Ultrasonic)

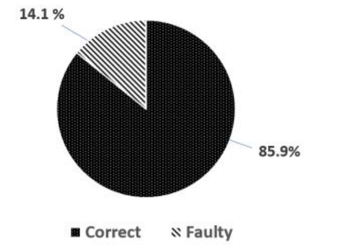

Figure 9. Left swipe observations (Ultrasonic)

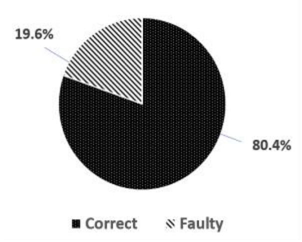

Figure 10. Scroll down observations (Ultrasonic)

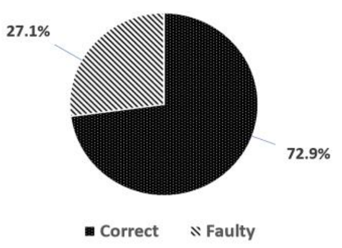

Figure 11. Scroll up observations (Ultrasonic)

Figure 12. Right swipe observations using barrier (Ultrasonic)

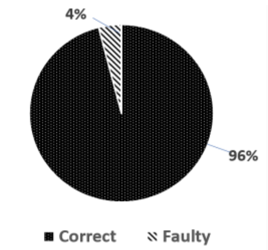

Figure 13. Left swipe observations (Infrared)

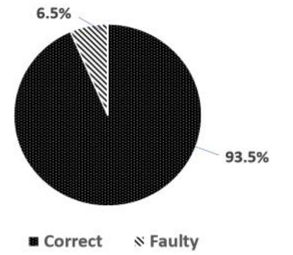

Figure 14. Right swipe observations (Infrared)

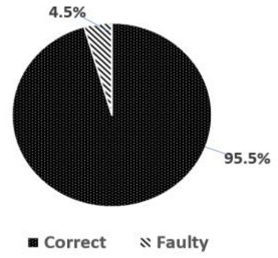

Figure 15. Scroll up observations (Infrared)

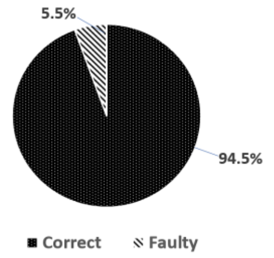

Figure 16. Scroll down observations (Infrared)

4. CONCLUSION

It has been quantified that IR sensors have a better accuracy in Gesture control technology than ultrasonic sensors. The increase in accuracy has been found out to about 27.874% through our experiment. Among the two technologies, complimenting the gesture motions, studied and compared thoroughly, error observations were noted, which were consolidated into the following conclusions: Cons in using ultrasonic sensors: Its range is up to 400 cm. If multiple sensors are put closely, there will be interference and it will give unwanted output which is unreliable. Moreover, if there is sound absorbing or reflecting material worn by the user or are in very close proximity of the sensors the readings will be faulty resulting in wrong commands given to the system. Cons in using IR sensors: It gives inaccurate reading if encountered with a reflective surface. If the user wears something that is reflective in nature or absorbs the beam of IR, the reading of the IR sensor can become inaccurate. If multiple sensors are put closely, there will be interference and it will give unreliable output. It can maintain high accuracy for a range of 150 cm after that irregularities increase in the readings from the sensor. Ultrasonic sensors work using the principle of sound waves. Their results can be affected due to many factors such as noise, other obstacle interactions, sensor liabilities. But can be used where there is less sound medium based noise and more heat related noise. Infrared, on the other hand, is a more accurate technology considering the gesture technologies and its application. It still has shortcomings; for instance, if there are many heat sources in the environment. Error arising in an ultrasonic sensor due to noise and other matters can be decreased by adding a barrier in between the ultrasonic sensor to decrease the interruption by the sound noises normal to the barrier and isolate the direction to be sensed.

ACKNOWLEDGEMENTS
This research work was funded by "Woosong University's Academic Research Funding-2022".

REFERENCES
[1] J. L. Raheja, R. Shyam, U. Kumar, and P. B. Prasad, "Real-time robotic hand control using hand gestures," in *2010 Second International Conference on Machine Learning and Computing*, 2010, pp. 12–16, doi: 10.1109/ICMLC.2010.12.
[2] S. Belgamwar and S. Agrawal, "An Arduino based gesture control system for human-computer interface," in *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, Aug. 2018, pp. 1–3, doi: 10.1109/ICCUBEA.2018.8697673.
[3] S. Jiang *et al.*, "Feasibility of wrist-worn, real-time hand, and surface gesture recognition via sEMG and IMU sensing," *IEEE Trans. Ind. Informatics*, vol. 14, no. 8, pp. 3376–3385, Aug. 2018, doi: 10.1109/TII.2017.2779814.
[4] J. P. Wachs, M. Kölsch, H. Stern, and Y. Edan, "Vision-based hand-gesture applications," *Commun. ACM*, vol. 54, no. 2, pp. 60–71, Feb. 2011, doi: 10.1145/1897816.1897838.
[5] C. Shan, "Gesture control for consumer electronics," in *Multimedia Interaction and Intelligent User Interfaces*, Springer London, 2010, pp. 107–128.
[6] S. Song, D. Yan, and Y. Xie, "Design of control system based on hand gesture recognition," in *2018 IEEE 15th International Conference on Networking, Sensing and Control (ICNSC)*, Mar. 2018, pp. 1–4, doi: 10.1109/ICNSC.2018.8361351.
[7] M. Sudarma, "Transformation of information technology based on human computer interaction concept," *Int. J. Informatics Commun. Technol.*, vol. 2, no. 3, Dec. 2013, doi: 10.11591/ij-ict.v2i3.4115.
[8] G. Sziladi, T. Ujbanyi, and J. Katona, "Cost-effective hand gesture computer control interface," in *2016 7th IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*, Oct. 2016, pp. 239–244, doi: 10.1109/CogInfoCom.2016.7804555.
[9] K. P., M. H. S., and V. P. A., "Human machine interface based on eye wink detection," *Int. J. Informatics Commun. Technol.*, vol. 2, no. 2, Jul. 2013, doi: 10.11591/ij-ict.v2i2.3927.
[10] M. Alam and M. A. Yousuf, "Designing and Implementation of a Wireless Gesture Controlled Robot for Disabled and Elderly People," in *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, Feb. 2019, pp. 1–6, doi: 10.1109/ECACE.2019.8679290.
[11] R. Runwal *et al.*, "Hand gesture control of computer features," in *Advances in Mechanical Engineering*, Springer Singapore, 2021, pp. 799–805.
[12] M. Jain, Aditi, A. Lohiya, M. F. Khan, and A. Maurya, "Wireless gesture control robot: an analysis," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 1, no. 10, 2012.
[13] D. Lee and Y. Park, "Vision-based remote control system by motion detection and open finger counting," *IEEE Trans. Consum. Electron.*, vol. 55, no. 4, pp. 2308–2313, Nov. 2009, doi: 10.1109/TCE.2009.5373803.
[14] P. K. Pisharady and M. Saerbeck, "Recent methods and databases in vision-based hand gesture recognition: A review," *Comput. Vis. Image Underst.*, vol. 141, pp. 152–165, Dec. 2015, doi: 10.1016/j.cviu.2015.08.004.
[15] Y. Han, "A low-cost visual motion data glove as an input device to interpret human hand gestures," *IEEE Trans. Consum. Electron.*, vol. 56, no. 2, pp. 501–509, May 2010, doi: 10.1109/TCE.2010.5505962.
[16] V. A. Shanthakumar, C. Peng, J. Hansberger, L. Cao, S. Meacham, and V. Blakely, "Design and evaluation of a hand gesture recognition approach for real-time interactions," *Multimed. Tools Appl.*, vol. 79, no. 25–26, pp. 17707–17730, Jul. 2020, doi: 10.1007/s11042-019-08520-1.
[17] M. Velliste, S. Perel, M. C. Spalding, A. S. Whitford, and A. B. Schwartz, "Cortical control of a prosthetic arm for self-feeding," *Nature*, vol. 453, no. 7198, pp. 1098–1101, Jun. 2008, doi: 10.1038/nature06996.
[18] Y. Zhu, Z. Yang, and B. Yuan, "Vision Based Hand Gesture Recognition," in *2013 International Conference on Service Sciences (ICSS)*, Apr. 2013, pp. 260–265, doi: 10.1109/ICSS.2013.40.
[19] M. Oudah, A. Al-Naji, and J. Chahl, "Hand Gesture Recognition Based on Computer Vision: A Review of Techniques," *J. Imaging*, vol. 6, no. 8, p. 73, Jul. 2020, doi: 10.3390/jimaging6080073.
[20] S. S. Rautaray, "Real Time Hand Gesture Recognition System for Dynamic Applications," *Int. J. UbiComp*, vol. 3, no. 1, pp. 21–31, Jan. 2012, doi: 10.5121/iju.2012.3103.
[21] M. S. Ghute, M. Anjum, and K. P. Kamble, "Gesture based mouse control," in *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, Mar. 2018, pp. 710–713, doi: 10.1109/ICECA.2018.8474905.
[22] N. I. A. M., N. Aini Z., S. R., and N. M. S., "Analysis of surface electromyography for hand gesure classification," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 15, no. 3, p. 1366, Sep. 2019, doi: 10.11591/ijeecs.v15.i3.pp1366-1373.
[23] J. R. Pansare, J. Shrotriya, R. Durrani, and S. Jaisinghani, "Real-time static hand gesture recognition system using HCI for recognition of numbers," *Int. J. Adv. Res. Comput. Sci.*, vol. 4, no. 4, 2013.
[24] A. A. Galadima, "Arduino as a learning tool," in *2014 11th International Conference on Electronics, Computer and Computation (ICECCO)*, Sep. 2014, pp. 1–4, doi: 10.1109/ICECCO.2014.6997577.
[25] S. Shreevidya, N. Namratha, V. M. Nisha, and M. Dakshayini, "Hand Gesture Based Human-Computer Interaction Using Arduino," 2020, pp. 315–321.
[26] J. S. Vimali, S. Srinivasulu, J. Jabez, and S. Gowri, "Hand gesture recognition control for computers using Arduino," in *Data Intelligence and Cognitive Informatics*, Springer Singapore, 2021, pp. 569–578.
[27] Y.-W. Bai, Z.-L. Xie, and Z.-H. Li, "Design and implementation of a home embedded surveillance system with ultra-low alert power," *IEEE Trans. Consum. Electron.*, vol. 57, no. 1, pp. 153–159, Feb. 2011, doi: 10.1109/TCE.2011.5735496.
[28] U. V. Solanki and N. H. Desai, "Hand gesture based remote control for home appliances: Handmote," in *2011 World Congress on Information and Communication Technologies*, Dec. 2011, pp. 419–423, doi: 10.1109/WICT.2011.6141282.
[29] K. B. Shaik, P. Ganesan, V. Kalist, B. S. Sathish, and J. M. M. Jenitha, "Comparative study of skin color detection and segmentation in HSV and YCbCr color space," *Procedia Comput. Sci.*, vol. 57, pp. 41–48, 2015, doi: 10.1016/j.procs.2015.07.362.
[30] A. K. H. AlSaedi and A. H. H. AlAsadi, "An efficient hand gestures recognition system," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 745, no. 1, p. 12045, Feb. 2020, doi: 10.1088/1757-899X/745/1/012045.

BIOGRAPHIES OF AUTHORS

Aditya Prasad Mahapatra is pursuing Bachelor of Technology (B.Tech) in the stream of Electrical and Electronics Engineering in International Institute of Information Technology Bhubaneswar (IIIT-BH), Odisha, India. His research interests include Internet of Things, Data Science and Machine Learning. He can be contacted by email: b318002@iiit-bh.ac.in.

Bishweashwar Sukla is pursuing Bachelor of Technology (B.Tech) in the stream of Electrical and Electronics Engineering in International Institute of Information Technology Bhubaneswar (IIIT-BH), Odisha, India. His research interests include Data science models, Deep Learning, Internet of Things. He can be contacted by email: b318010@iiit-bh.ac.in.

Harikrishnan KM is pursuing Bachelor of Technology (B.Tech) in the stream of Electrical and Electronics Engineering in International Institute of Information Technology Bhubaneswar (IIIT-BH), Odisha, India. His research interests include Data science models, Gesture control technologies, Hardware robotic technologies and Big Data. He can be contacted by email: b318017@iiit-bh.ac.in.

Debani Prasad Mishra received the B.Tech. in electrical engineering from the Biju Patnaik University of Technology, Odisha, India, in 2006 and the M.Tech in power systems from IIT, Delhi, India in 2010. He has been awarded the Ph.D. degree in power systems from Veer Surendra Sai University of Technology, Odisha, India, in 2019. He is currently serving as Assistant Professor in the Dept of Electrical Engg, International Institute of Information Technology Bhubaneswar, Odisha. He has 11 years of teaching experience and 2 years of industry experience in the thermal power plant. He is the author of more than 80 research articles. His research interests include soft Computing techniques application in power systems, signal processing and power quality. 3 students have been awarded Ph.D. under his guidance and currently 4 Ph.D. Scholars are continuing under him. He can be contacted at email: debani@iiit-bh.ac.in.

Surender Reddy Salkuti received the Ph.D. degree in electrical engineering from the Indian Institute of Technology, New Delhi, India, in 2013. He was a Postdoctoral Researcher with Howard University, Washington, DC, USA, from 2013 to 2014. He is currently an Associate Professor with the Department of Railroad and Electrical Engineering, Woosong University, Daejeon, South Korea. His current research interests include power system restructuring issues, ancillary service pricing, real and reactive power pricing, congestion management, and market clearing, including renewable energy sources, demand response, smart grid development with integration of wind and solar photovoltaic energy sources, artificial intelligence applications in power systems, and power system analysis and optimization. He can be contacted at email: surender@wsu.ac.kr.