JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

REAL-TIME HAND GESTURE DETECTION AND RECOGNITION SYSTEM FOR HUMAN-COMPUTER INTERACTION

[1]L.L.Limla Shanoo, [2]S.Kirubavilashini, [3]M.Asha, [4]K.Priya, [5]P.Malini,
[1,2, 3,4] BE Students, Department of Electronics and Communication Engineering,
[5]Assistant Professor, Department of Electronics and Communication Engineering,
Vivekanandha College of Technology for Women, Elayampalayam, Tiruchengode, Namakkal Dt-637205

Abstract : Using a physical device for human-computer interaction, such as a mouse or keyboard, interferes with the natural interface by creating a strong barrier between the user and the computer. In this project, we created a powerful marker-less hand gesture detection system to efficiently monitor static hand motions. With the fast growth of technology, home automation systems are gaining popularity. Amazon Alexa and Google Home have grabbed the market by storm. A Smart Home System provides access to all controlled household equipment and can assist in improving the electrical supply in a household, lowering consumption by managing utilization. Existing developments in this sector depend mostly on sound. This research introduces a system that uses Virtual Reality gestures to control household equipment. The suggested system utilizes an image analysis technique to operate VR-based smart home devices. Additionally, the system offered a gesture-based multipurpose, low-cost, and adaptable system. The output prototype using image processing technology from VR technology is more appropriate for smart and household appliances.

Keywords **- Human-Computer Interaction, Virtual Reality Gestures, Image Processing Technology.**

I. INTRODUCTION

The Web Camera-Based Home Automation System with Hand Gestures for Light and Fan Control aims to transform how people interact with their homes. The device seeks to give exceptional ease and simplicity by utilizing hand gesture detection technology. Users may operate their home's lighting and fan systems with simple hand gestures, eliminating the need for traditional physical switches or remote devices. Furthermore, the system seeks to promote accessibility by offering an alternative form of contact for persons with mobility restrictions, assuring equality for all users. By using virtual reality technology, the system intends to take the user experience to new heights, allowing users to immerse themselves in a virtual recreation of their home environment while effortlessly manipulating appliances with their movements. With a focus on reliability, safety, scalability, and flexibility, the objective is to create a home automation system that not only meets but surpasses current inhabitants' expectations, bringing in a new era of smart living.

II. DESCRIPTION

2.1 AIM

The project aims to develop an Arduino Nano-based system for detecting and recognizing hand gestures and controlling certain applications.

2.2 OBJECTIVES

a. To build the circuit that should understand the gestures accurately through the web camera.
b. To do programming in python using Visual Studio code for recognizing the hand gestures.
c. To permit the user, to intuitively operate appliances especially people with disabilities.

2.3 METHODOLOGY

In this project, we create a gesture control laptop using Arduino Nano and Python. Besides employing input devices such as a keyboard, mouse, or joystick, hand gestures are used to offer input to the laptop or PC. The essential premise of our project is that

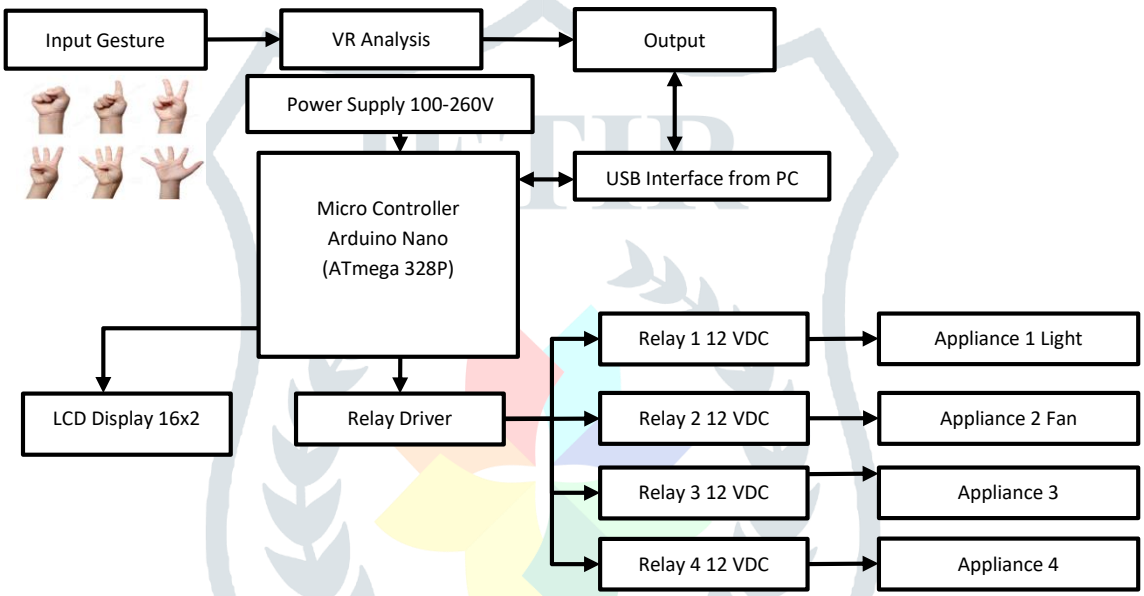
inputs are supplied by hand gestures, which are recognized by the web camera and sent to the Arduino Nano, after which the microcontroller transmits orders to an LCD module that displays pertinent information. This information might contain the recognized gesture, the system status, or any other pertinent information. The microprocessor manages the relay driver circuit. The relay driver turns on or off the appliances in response to the microcontroller's orders.

Programming in Arduino:

Arduino programming involves receiving gesture data from the PC, processing it into commands, and calibrating it to do certain tasks. The input, which is a hand motion in front of the webcam, is identified and then calibrated to conduct tasks.

Programming in Python:

To program in Python, we utilized two libraries. The NumPy library provides precompiled logical and numerical bounds, and its capabilities ensure fast execution. and OpenCV library, which is meant to address PC vision difficulties.



2.4 REQUIREMENTS:

Hardware	Software
1. Arduino Nano board 2. LCD Display 3. Jumper wires 4. USB cable 5. Laptop or PC 6. Lamp 7. DC Fan 8. Relay and Relay Driver Circuit	1. Arduino IDE 2. Python IDE 3.7 3. Visual Studio Code

III. SOFTWARE DESIGN & SIMULATION

3.1 CIRCUIT DIAGRAM

The circuit schematic for the Arduino section of the project is shown in the following image. It consists of an Arduino Nano board, four relays, a DC fan, and LED, all of which can be powered by SMPS Power Supply and controlled via the laptop's USB port.

3.2 CIRCUIT DESCRIPTION:

S.No	Description	Specification
1	Microcontroller	ATmega328P
2	Architecture	AVR
3	Operating Voltage	5 Volts
4	Flash Memory	32 KB
5	SRAM	2 KB
6	Clock Speed	16 MHz
7	Analog I/O Pins	8
8	EEPROM	1KB
9	DC Current per I/O Pins	40 milli/Amps
10	Input Voltage	7-12 Volts

3.3 SOFTWARE SIMULATION:

We employed two software for software simulation: Arduino IDE, an integrated programming environment available on Windows, Mac OS, and Linux, and functions written in C and C++. The Arduino IDE is used to write and upload code to Arduino-compatible boards. The second software we use is Python IDE, which offers default language implementations.

We use Python as a programming language to create our appliances. OpenCV is an excellent image-processing tool, and it is widely used as a Python library of real-time computer vision technologies. This open-source framework can monitor complaints, recognize people, locate places, and do much more. It is interoperable with several programming languages, including Python, Java, and C++.

3.4 INSTALLING LIBRARIES:

a) Open CV

OpenCV is a Python package designed to address PC vision challenges. It supports a variety of languages and has been the most essential library for dealing with recognition-type applications in which a frame or video must be taken and processed, whether for object detection or motion detection. OpenCV is a Python library that is very useful in calculating hand tracing. It is equipped with certain functions that are the steps toward creating a model with signal identification. These functions include video capturing, background removal, image editing (resizing, rotation), plotting a motion detection graph, and so on.

b) Numpy

NumPy is a Python package. The word NumPy refers to Numerical Python and is used. It is a Python extension module, often written in C. This ensures that the precompiled logical and numerical limitations, as well as the Numpy features, provide outstanding execution speed. It is used to easily do complex computations such as dot product and summation. It is a very efficient module that makes work easier.

3.5 SIMULATION

a) Module of Camera

The camera module is used to take input from multiple image detectors and then communicate it to the detection module for additional preprocessing. Some techniques for capturing input data are available on the market. These include data gloves and cameras.In our experiment, we employed a built-in webcam camera that is inexpensive and can identify static motions quickly. USB-based cameras are also available at a premium price.

b) Module of detection.

The information collected from the camera module is processed in a variety of processes, including shade modification, noise removal, frequency evolution, extraction of various RGB outlines, and so on. This may result in two situations: an image with a defect and an image without a defect. If the signal is dynamic, then outlines with five consistent developments may be the most crucial aspect.

Using this strategy, we get the following result: the discovery isn't exceptionally precise; we tried specific qualities for edge, but the results were not exact; this could be because of the interaction of shadow, which significantly changes the foundation; additionally, the entire hand isn't in motion, making it difficult to recognize it; thus, we chose not to use this technique in the product.

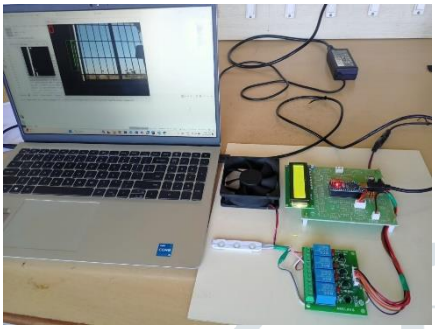
c) Module of interface

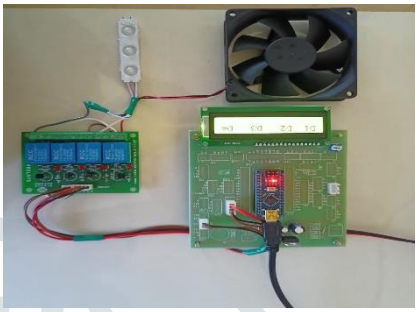
The activities must be transferred to the appropriate application. This module is in charge of mapping the identified hand gestures to the irrelated activities. The forward part comprises of 3windows. The main window comprises the video input that is caught from the camera with the related name of the motion recognized. The second shows the forms found inside the information pictures. What's more, the first window shows the smooth threshold variant of the picture. The edge and form window are

components of the Graphical UI; by including them, the client is aware of foundation irregularities that may affect the contribution to the framework, and they can adjust their workstation or work area web camera to keep a strategic distance from them. This would lead to improved execution.

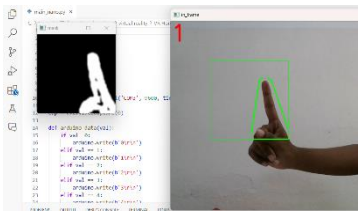
IV RESULTS AND DISCUSSION

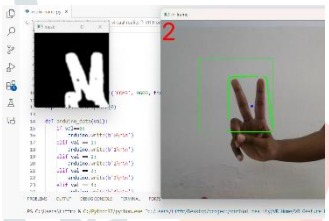
4.1 HARDWARE SETUP SNAPSHOTS

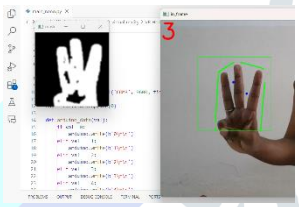


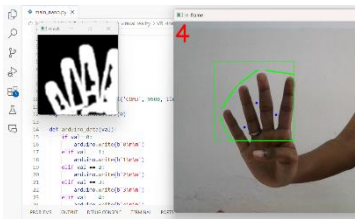


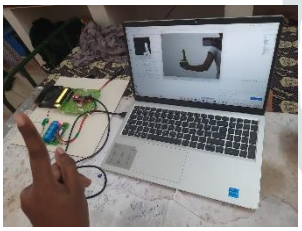
4.2 RESULT SNAPSHOTS:

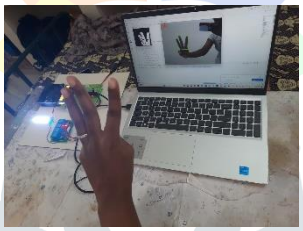












This project identified the count of fingers depicted in the illustration. Our original attempt to develop a gesture detection system was based on background removal. Many challenges and accuracy concerns arose when constructing the system with background subtraction. Background subtraction cannot deal with abrupt, severe illumination changes, resulting in several discrepancies. When employed against a simple background, the gesture detection technology proved to be durable and accurate. This precision was maintained regardless of the hue of the backdrop, as long as it was plain, solid, and free of irregularities. In circumstances when the backdrop was not plain, the objects inside it were determined to represent inconsistencies in the picture capture process, resulting in inaccurate results. As a result, it is advised that this system be utilized with a clean background to provide the simplest possible results and high accuracy.

V CONCLUSION

The combination of an image analysis system and VR technology improves the goal and value of the use and maintenance management of construction equipment, ensures the safe and stable operation, efficient operation, and energy-saving operation of construction equipment, and ensures the construction of the entire construction life cycle. The application value of information models and virtual reality technologies. Simultaneously, the virtual simulation training system is utilized to employ virtual architecture, analyze the data, generate quantitative training indicators and appropriate training programs, and make improved suggestions for suitable elderly homes. Furthermore, the big data module always detects flaws in the training system platform and other functional modules by analyzing and integrating diverse data and then optimizes these modules and platforms to update and improve the complete system.

VI FUTURE WORK

The system is presently only tested on static gesture photos, but it may be expanded to identify dynamic motions in movies in real-time. The system may be customized to identify RGBD pictures captured by Kinetic Sensors. The article may be developed further by including deep learning approaches such as modified convolutional neural networks, as well as improving feature selection utilizing quantum computing and evolutionary algorithms following feature extraction. The model may be trained using physical hand models with sensors, which uses graph theory to offer additional data that can be evaluated to improve accuracy.

VII ACKNOWLEDGMENT

The Author would like to thank the project Guide, Prof. P. Malini who helped us in our project.

REFERENCES

[1] Sajjad Ur Rahman, Zeenat Afroze, Mohammed Tareq, "Hand Gesture Recognition Techniques For Human Computer Interaction Using OpenCV", International Journal of Scientific and Research Publications, Volume 4, Issue 12, December 2014
[2] M. Panwar, "Hand Gesture Recognition based on Shape Parameters," 2012 International Conference on Computing, Communication and Applications, pp.1-6, Feb. 2012.
Aditya, Arun Ramamurthy, et al (2014), 'Recognition of dynamic hand gestures', International Journal of Recent Technology and Engineering (IJRTE), Vol.11, No.10,pp. 14-34.
[3] Rishabh Agrawal and Nikita Gupta published in IEEE.(2010) "Real Time Hand Gesture Recognition for Human-Computer Interaction"
[4] U. Rajkanna, M. Mathankumar, and K. Gunasekaran published in IEEE "Hand gesture-based computer control using microcontroller" on 7-8 Jan 2012, Rohtak, Haryana, India.
[5] Seonghun Park, Ho-Seung Cha, and Chang-Hwan Im, "Development of an Online Home Appliance Control System Using Augmented Reality and an SSVEP-Based Brain-Computer Interface", IEEE Access, vol. 7, pp. - 163604 – 163614, 2019