CONTENTS

[1 INTRODUCTION 1](#_Toc385319430)

[1.1 Abstract 1](#_Toc385319431)

[1.2 Introduction and Motivation 1](#_Toc385319432)

[1.3 Problem Statement 2](#_Toc385319433)

[1.4 Scope of the project 2](#_Toc385319434)

[1.5 Organization of Project Report 2](#_Toc385319435)

[2 REVIEW OF LITERATURE 4](#_Toc385319436)

[2.1 State of the Current Methodology and Technology used 4](#_Toc385319437)

[2.2 Methodology and technology used (in Project) 5](#_Toc385319438)

[3 ANALYSIS AND DESIGN 10](#_Toc385319439)

[3.1 Requirement Analysis 10](#_Toc385319440)

[3.2 Project Design 14](#_Toc385319441)

[4 IMPLEMENTATION AND RESULTS 21](#_Toc385319442)

[4.1 Implementation Details 21](#_Toc385319443)

[4.2 Results 21](#_Toc385319444)

[5 TESTING 22](#_Toc385319445)

[5.1 Test Cases 22](#_Toc385319446)

[6 CONCLUSION AND FURTHER WORK 23](#_Toc385319447)

[6.1 Conclusion 23](#_Toc385319448)

[6.2 Further Work 23](#_Toc385319449)

# 1 INTRODUCTION

## 1.1 Abstract

Gesture Control is a subject which has been investigated almost from the beginning of using terminals to interact with the computer central unit. The advent of Kinect has sparked a series of efforts to apply gesture control not only in gaming, but rather in controlling TVs, PCs, laptops, smart home devices etc. A smart home needs to control their environments such as its temperature, humidity, light condition, and others in an intelligent manner. One simple way of controlling their environments is using the human gestures. A gesture is a movement that we make with a part of our body, face and hands as an expression of meaning or intention. The system interprets the user hand gestures into pre-defined commands to control smart home environments such as lights, fans and other electronic devices. We apply the proposed simultaneous gesture segmentation and recognition method to recognize the hand gestures for controlling the electronic devices in a smart home environment.

## 1.2 Introduction and Motivation

Gesture recognition is a topic in [computer science](http://en.wikipedia.org/wiki/Computer_science) and [language technology](http://en.wikipedia.org/wiki/Language_technology) with the goal of interpreting human [gestures](http://en.wikipedia.org/wiki/Gesture) via mathematical [algorithms](http://en.wikipedia.org/wiki/Algorithms). Gestures can originate from any bodily motion or state but commonly originate from the [face](http://en.wikipedia.org/wiki/Face) or [hand](http://en.wikipedia.org/wiki/Hand). Current focuses in the field include emotion recognition from the face and hand gesture recognition. Many approaches have been made using cameras and [computer vision](http://en.wikipedia.org/wiki/Computer_vision) algorithms to interpret [sign language](http://en.wikipedia.org/wiki/Sign_language). However, the identification and recognition of posture, gait, [proxemics](http://en.wikipedia.org/wiki/Proxemics), and human behaviors is also the subject of gesture recognition techniques.

Gesture recognition can be seen as a way for computers to begin to understand human [body language](http://en.wikipedia.org/wiki/Body_language), thus building a richer bridge between machines and humans than primitive [text user interfaces](http://en.wikipedia.org/wiki/Text_user_interface) or even [GUIs](http://en.wikipedia.org/wiki/GUI) (graphical user interfaces), which still limit the majority of input to keyboard and mouse. Gesture recognition can be conducted with techniques from [computer vision](http://en.wikipedia.org/wiki/Computer_vision) and [image processing](http://en.wikipedia.org/wiki/Image_processing).

Over the past few years, gesture recognition has made its debut in entertainment and gaming markets. Now, gesture recognition is becoming a commonplace technology, enabling humans and machines to interface more easily in the home, the automobile and at work. Imagine a person sitting on a couch, controlling the lights and fan with a wave of his hand. This and other capabilities are being realized as gesture recognition technologies which enable natural interactions with the electronics that surround us. Gesture recognition has long been researched with 2D vision, but with the advent of 3D sensor technology, its applications are now more diverse, spanning a variety of markets.

## 1.3 Problem Statement

The aim of the project is to develop a system for home automation using mere gestures. The project will observe an automated light system being controlled by using gestures. Each gesture will have corresponding action associated with, to lit the light or turn it off and to switch on the fan or switch it off. At least four gestures will be recognized by the system and corresponding actions will be performed.

## 1.4 Scope of the project

The project would be implemented in two phases:

1. Software implementation: Using various image processing algorithms, the acquired image would be processed upon to identify the predefined gestures.

2. Hardware implementation: Every recognized image would have a corresponding value which would be transmitted using wireless technology-Bluetooth. The inbuilt Bluetooth of the laptop can be used to achieve this or by connecting external Bluetooth dongle. On the receiver end, the Bluetooth signal will be interpreted by the integrated Bluetooth on the Arduino microcontroller and the corresponding actions would be taken by the data sent by the laptop on the receiver side.

## 1.5 Organization of Project Report

The project report consists of the following chapters.

**1.5.1 Review of Literature**

Computer vision is a field that includes methods for acquiring, processing, analyzing, and understanding images and, in general, high-dimensional data from the real world in order to produce numerical or symbolic information, *e.g.*, in the forms of decisions. Application building involves processing in various technologies. It involves image processing and processing in other fields to build up applications. The proposed system works on real time environment with in corporation of home appliance control using arduino circuitry, everything running simultaneously making it a complete system using minimal resources. One can control various devices using pre-defined gestures. The system now controls lights and fans using the gesture recognition using Bluetooth as a source to communicate.

**1.5.2 Analysis and Design**

The proposed system takes pre-defined gesture as input and performs the assigned tasks. There are many steps through which the whole system goes to perform a complete task. Comparison of many techniques of gesture recognition is performed. Analysis based on the different algorithm is presented. Various techniques are used to identify the gesture. There on the signal is communicated using Bluetooth for controlling devices.

**1.5.3 Implementation and Results**

The code implementation involves gesture recognition and the assigned task is performed. The user needs to place his/her hand in front of the system camera and show the pre-defined gesture for the task it needs to perform. The system identifies the gesture and action is completed. The system being dynamic, u need to change the gesture to perform the corresponding action.

**1.5.4 Testing**

The system is tested with different techniques and all the pre-defined gestures to get the desired actions preformed.

**1.5.5 Conclusion**

Thus, the hand gesture recognition system is built and tested on the inputs and it successfully yields desired results.

# 2 REVIEW OF LITERATURE

## 2.1 State of the Current Methodology and Technology used

Gestures can originate from any bodily motion or state but commonly originate from the face or hand. Current focuses in the field include emotion recognition from the face and hand gesture recognition. Many approaches have been made using cameras and computer vision algorithms to interpret sign language. However, the identification and recognition of posture, gait, proxemics, and human behaviors is also the subject of gesture recognition techniques. Gesture recognition can be seen as a way for computers to begin to understand human body language, thus building a richer bridge between machines and humans than primitive interfaces or even GUIs (graphical user interfaces), which still limit the majority of input to keyboard and mouse.

Gesture recognition enables humans to communicate with the machine (HIM) and interact naturally without any mechanical devices. Using the concept of gesture recognition, it is possible to point a finger at the computer screen so that the cursor will move accordingly. This could potentially make conventional input devices such as mouse, keyboards and even touch-screens redundant.

Gesture recognition can be conducted with techniques from computer vision and image processing.

Kinect is a [motion sensing](http://en.wikipedia.org/wiki/Motion_sensing) [input device](http://en.wikipedia.org/wiki/Input_device) by [Microsoft](http://en.wikipedia.org/wiki/Microsoft) for the [Xbox 360](http://en.wikipedia.org/wiki/Xbox_360) [video game console](http://en.wikipedia.org/wiki/Video_game_console) and [Windows](http://en.wikipedia.org/wiki/Windows) [PCs](http://en.wikipedia.org/wiki/Personal_computer). Based around a [webcam](http://en.wikipedia.org/wiki/Webcam)-style add-on [peripheral](http://en.wikipedia.org/wiki/Peripheral) for the Xbox 360 console, it enables users to control and interact with the Xbox 360 without the need to touch a [game controller](http://en.wikipedia.org/wiki/Game_controller), through a [natural user interface](http://en.wikipedia.org/wiki/Natural_user_interface) using gestures and [spoken commands](http://en.wikipedia.org/wiki/Speech_recognition). The project is aimed at broadening the Xbox 360's audience beyond its typical gamer base. Kinect competes with the [Wii Remote Plus](http://en.wikipedia.org/wiki/Wii_Remote_Plus) and [PlayStation Move](http://en.wikipedia.org/wiki/PlayStation_Move) with [PlayStation Eye](http://en.wikipedia.org/wiki/PlayStation_Eye) [motion controllers](http://en.wikipedia.org/wiki/Motion_controller) for the [Wii](http://en.wikipedia.org/wiki/Wii) and [PlayStation 3](http://en.wikipedia.org/wiki/PlayStation_3) home consoles, respectively. A version for [Windows](http://en.wikipedia.org/wiki/Microsoft_Windows) was released on February 1, 2012.

Kinect was launched in North America on November 4, 2010, in Europe on November 10, 2010, in Australia, New Zealand and Singapore on November 18, 2010, and in Japan on November 20, 2010. Purchase options for the sensor peripheral include a bundle with the game Kinect and console bundles with either a 4 GB or 250 GB Xbox 360 console and *Kinect Adventures*.

The Kinect claimed the [Guinness World Record](http://en.wikipedia.org/wiki/Guinness_World_Record) of being the "fastest selling consumer electronics device" after selling a total of 8 million units in its first 60 days. 24 million units of the Kinect sensor had been shipped as of January 2012.

Microsoft released Kinect [software development kit](http://en.wikipedia.org/wiki/Software_development_kit) for Windows 7 on June 16, 2011. This SDK was meant to allow developers to write Kinecting apps in [C++/CLI](http://en.wikipedia.org/wiki/C%2B%2B/CLI), [C#](http://en.wikipedia.org/wiki/C_Sharp_(programming_language)), or [Visual Basic .NET](http://en.wikipedia.org/wiki/Visual_Basic_.NET).

## 2.2 Methodology and technology used (in Project)

**2.2.1 Image Processing:**

In [imaging science](http://en.wikipedia.org/wiki/Imaging_science), image processing is any form of [signal processing](http://en.wikipedia.org/wiki/Signal_processing) for which the input is an image, such as a [photograph](http://en.wikipedia.org/wiki/Photograph) or [video frame](http://en.wikipedia.org/wiki/Video_frame); the [output](http://en.wikipedia.org/wiki/Output) of image processing may be either an image or a set of characteristics or [parameters](http://en.wikipedia.org/wiki/Parameter) related to the image. Most image-processing techniques involve treating the image as a [two-dimensional](http://en.wikipedia.org/wiki/Two-dimensional) [signal](http://en.wikipedia.org/wiki/Signal_(electrical_engineering)) and applying standard signal-processing techniques to it.

Image processing usually refers to [digital image processing](http://en.wikipedia.org/wiki/Digital_image_processing), but [optical](http://en.wikipedia.org/wiki/Optical_engineering) and [analog image processing](http://en.wikipedia.org/wiki/Analog_image_processing) also are possible. This article is about general techniques that apply to all of them. The acquisition of images (producing the input image in the first place) is referred to as [imaging](http://en.wikipedia.org/wiki/Imaging_science).

Modern digital technology has made it possible to manipulate multi-dimensional signals with systems that range from simple digital circuits to advanced parallel computers. The goal of this manipulation can be divided into three categories:

* Image Processing (image in -> image out)
* Image Analysis (image in -> measurements out)
* Image Understanding (image in -> high-level description out)

Most usually, image processing systems require that the images be available in digitized form, that is, arrays of finite length binary words. For digitization, the given Image is sampled on a discrete grid and each sample or pixel is quantized using a finite number of bits. The digitized image is processed by a computer. To display a digital image, it is first converted into analog signal, which is scanned onto a display.

**2.2.2 Computer Vision:**

Computer vision is a field that includes methods for acquiring, [processing](http://en.wikipedia.org/wiki/Digital_image_processing), analyzing, and understanding images and, in general, high-dimensional data from the real world in order to produce numerical or symbolic information, *e.g.*, in the forms of decisions. A theme in the development of this field has been to duplicate the abilities of human vision by electronically perceiving and understanding an image. This image understanding can be seen as the disentangling of symbolic information from image data using models constructed with the aid of geometry, physics, statistics, and learning theory. Computer vision has also been described as the enterprise of automating and integrating a wide range of processes and representations for vision perception.

Applications range from tasks such as industrial [machine vision](http://en.wikipedia.org/wiki/Machine_vision) systems which, say, inspect bottles speeding by on a production line, to research into artificial intelligence and computers or robots that can comprehend the world around them. The computer vision and machine vision fields have significant overlap. Computer vision covers the core technology of automated image analysis which is used in many fields. Machine vision usually refers to a process of combining automated image analysis with other methods and technologies to provide automated inspection and robot guidance in industrial applications.

As a scientific discipline, computer vision is concerned with the theory behind artificial systems that extract information from images. The image data can take many forms, such as video sequences, views from multiple cameras, or multi-dimensional data from a medical scanner.

Computer vision is concerned with modeling and replicating human vision using computer software and hardware. It combines knowledge in computer science, electrical engineering, mathematics, physiology, biology, and cognitive science. It needs knowledge from all these ﬁelds in order to understand and simulate the operation of the human vision system.

**2.2.3 Arduino Interfacing and Wireless Transmission:**

**Wireless communication** is among technology’s biggest contributions to mankind. Wireless communication involves the transmission of information over a distance without help of wires, cables or any other forms of electrical conductors. The transmitted distance can be anywhere between a few meters (for example, a television’s remote control) and thousands of kilometers (for example, radio communication). The wireless technology used in the project is Bluetooth. It is paired with the Bluetooth on the laptop and hence the recognized gestured has been assigned with some character and this character is transmitted using serial communication via Bluetooth. The data which has been received is then checked and the corresponding action is performed.

**2.2.4 Hardware and Software Required**

**2.2.4.1 OpenCV 2.4.8 and Visual Studio (C++) 2013**

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being a BSD-licensed product, OpenCV makes it easy for businesses to utilize and modify the code.

The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces, identify objects, classify human actions in videos, track camera movements, track moving objects, extract 3D models of objects, produce 3D point clouds from stereo cameras, stitch images together to produce a high resolution image of an entire scene, find similar images from an image database, remove red eyes from images taken using flash, follow eye movements, recognize scenery and establish markers to overlay it with augmented reality, etc.

There are over 500 algorithms and about 10 times as many functions that compose or support those algorithms. OpenCV is written natively in C++ and has a template interface that works seamlessly with STL containers.

OpenCV was designed to be cross-platform. So, the library was written in C and this makes OpenCV portable to almost any commercial system. Since version 2.0, OpenCV includes its traditional C interface as well as the new C++ one. For the most part, new OpenCV algorithms are now developed in C++.

Microsoft Visual Studio is an [integrated development environment](http://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) from [Microsoft](http://en.wikipedia.org/wiki/Microsoft). Visual Studio supports different [programming languages](http://en.wikipedia.org/wiki/Programming_language) by means of language services, which allow the code editor and debugger to support (to varying degrees) nearly any programming language, provided a language-specific service exists. Built-in languages include [C](http://en.wikipedia.org/wiki/C_(programming_language))/[C++](http://en.wikipedia.org/wiki/C%2B%2B) (via [Visual C++](http://en.wikipedia.org/wiki/Visual_C%2B%2B)).

**2.2.4.2 Arduino Interfacing**

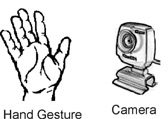
Once the gesture has been recognized the processing shifts to the arduino board for transmission and reception of the Bluetooth signal for controlling the electronic devices. The micro-controller board is programmed using Arduino Programming IDE. The Bluetooth dongle is connected to the arduino board at the receiving end. The signal to be transmitted is sent using the in-built Bluetooth of the laptop. At the receiver end the signal received is then interpreted and the corresponding action is performed.

**2.2.4.3 Bluetooth Socket Programming**

In order to send the corresponding data assigned to an image, we need to invoke the Bluetooth connected to the laptop. Since the in-built Bluetooth of the laptop is used, the Bluetooth of the laptop is first paired with the Bluetooth modem at the receiver’s side. Then Bluetooth Socket is created in order to start communication of the in-built Bluetooth of the laptop and the External Bluetooth transceiver.

**2.3 Project Overview**

Lights On



Lights Off

Fan On

Transmission of Signal

Hardware Interfacing Circuit

Hand Gesture Processing Unit

Fan Off

Fig 2.1: Project Overview

# 3 ANALYSIS AND DESIGN

## 3.1 Requirement Analysis

**3.1.1 Functional Requirements**

1. Webcam detection and connection: The system should be able to detect and connect to webcam in order to detect hand motion and gestures.

2. Hand Tracking: The system should be able to process the images provided by the webcam, detect the user hand position and notify OpenCV

3. Gesture recognition: The system should be able to process the input stream and identify the hand gestures

4. Simultaneous Execution: The system should be able to run gesture recognition and hand tracking simultaneously allowing each capability to interact with the other

5. Lighting: The lighting devices should be set up in such a way that the changes in ambient illumination such as sun light changing with the weather or the time of the day should not compromise image analysis and processing. There should be proper backlighting so that the edges and distances can be detected properly minimizing the errors.

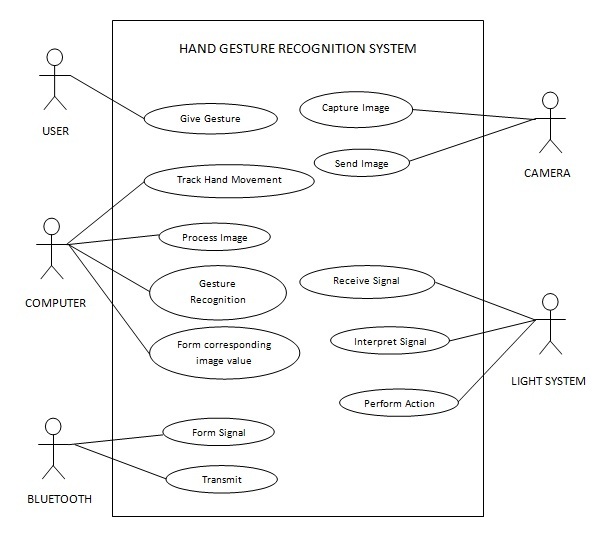


Fig 3.1: Use case diagram for Hand Gesture Recognition System

The activity diagram for the image processing can be shown below:

Switch on webcam

Detect skin color

No

Is it palm region?

Yes

Draw contour & convex hull

Process Image

Fig 3.2: Activity diagram

|  |
| --- |
| Camera |
| Detect image |
| Capture image |
| Store image |

The class diagram for Gesture Recognition for Electronics Control System can be shown below:

Gesture Recognition

Image enhancement

Image recognition

Image value

|  |
| --- |
| User |
| Name |
| Age |

Light System

Receive value

Perform action

Bluetooth

Transmit signals

Fig 3.3: Class diagram

**3.1.2 Non-functional Requirements**

1. Easy access: The system should be accessible to the single end user and should not produce any type of complexity during the use

2. Accuracy: The system should be as accurate as possible to always provide exact result to the appropriate user

3. Performance: There should not be much delay between the input and the performed action. Processing of inputs is very important, so from the performance point of view movement detection algorithms must be as efficient as possible.

4. Efficiency: The amount of input is not very huge but it comes repeatedly and application must process them all with respect to an algorithmic sequence.

5. Usability: The usability of the software depends on the following factors:

* The systems must be robust yet flexible enough to cater for different images of the same person in different states of mind as the images can be a little different.
* Users must know how to interact with the hardware.

6. Reliability: There are always variations in the images of the same user. The reliability of the system depends on the fact that the system should be flexible enough to handle small variations in the images of the user and the system should be robust enough to identify the correct gesture.

7. Extensibility: The system should be such that the implementation takes into consideration future growth.

## 3.2 Project Design

**3.2.1 Design Consideration**

A simple approach is followed while designing and consists of the steps in order:

1. Image Acquisition
2. Pre-processing
3. Gesture Detection
4. Classification
5. Creating Bluetooth Socket
6. Action based on detected gesture

**3.2.2 Design Details**

The operation of the system proceeds in following basic steps:

1. Image input
2. Skin Segmentation
3. Image Processing and data extraction
4. Hand Gesture Recognition
5. Sending appropriate signal to hardware
6. Action performed

**3.2.2.1 Image Input**

To input image data into the system, a video can be used with an image capture program used to take the picture. The camera captures frames of videos on an auto run program. The image captured includes the hand stretched out and also the background.

**3.2.2.2 Skin Segmentation**

To extract the skin elements from the image, the image is converted from RGB to HSV format. It helps in recognizing the skin elements in the image.

**3.2.2.3 Image Processing and data extraction**

After the skin segmentation is been done, the image includes noise and the image is not smooth enough for further process. Here the noise is eliminated and the output of this stage is a smooth region of the hand figure, which is stored in a logical bitmap image.

**3.2.2.4 Hand Gesture Recognition and Tracking**

**3.2.2.4.1 Thresholding**

Having extracted the moving object region, the thresholding on the frame difference can be applied to extract the possible moving region in complex background. Conventional thresholding methods, such as Ostu thresholding, are not suitable for the case of detecting motion difference. Instead, a simple thresholding technique can be used to extract moving regions. The thresholding formation detection is determined.

**3.2.2.4.2 Skin color detection**

Skin can be easily detected by using skin segmentation. A threshold filter is applied to remove the non-skin elements. The major advantage of this approach is that the influence of luminosity can be removed during the conversion process. Thus it makes the segmentation less dependent on the lightning condition, which has always been a critical obstacle for image recognition. The threshold values used in our data set are:

Hue (H): 0-20

Saturation (S): 40-150

Value (V): 60-255

This values range according to different surroundings and light conditions. This range can be determined in any surrounding you need.

**3.2.2.4.3 Edge Detection**

Edge detection is applied to separate the arm region from the hand region. It is easy to find the hand region. It is easy to find that there are fewer edges on the arm region than on the palm region. Here, we use a simple edge detection technique (e.g. Kirsch edge operator) to obtain different direction edges, and then choose the absolute maximum value of each pixel to form the edge image of respective frame.

**3.2.2.4.4 Combination of motion, skin color and edge detection**

The hand gestures information consists of movement, skin color and edge feature. We use the logic 'AND' to combine these three types of information. The combined image has many features that can be extracted because the different image processing methods have extracted different kind of information. Each image consists of different characteristic regions such as motion region, skin color regions and edge regions. The combined image consists of large region in the palm area and some small regions in the arm area. These two regions are separated to allocate the hand region.

**3.2.2.4.5 Region Identification**

A simple method for region identification is to label each region with a unique integer number which is called the labeling process. After labeling, the largest integer label indicates the number of regions in the image. After the labeling process, the small regions can be treated as noise and then be removed.

**3.2.2.4.6 Hand Tracking**

**3.2.2.4.6.1 Hand Localization**

Through an incoming frame from a real time video, the main issue is localizing the hand region accurately. In the first frame, the system obtains the complete color image that needs to be segmented and converted into HSV color model. The randomized lists-based classifier takes the Hue, Saturation and Value elements of each pixel. Randomized lists directly label the pixels belonging to the hand. The pixel with the value 1 as the output represents the hand segmentation results. Once the pixels are labeled different techniques are applied to group the labeled pixels into several connected regions. During this stage, different regions need to be segmented which consist of a region from the background similar to hand color distribution. The hand region retains the labeled pixel with largest number in the region. Then a feature point defined as interaction point will be extracted which will be used as an input device. In the present implemented technique, apply the center of the tracked hand region. This interaction can be used as an input for interactive application. By next incoming frame, the system obtains abounded region that circumferences the tracked hand in previous frame, and further repeats the same procedure to segment the hand, thus making the segmentation faster.

**3.2.2.4.6.2 Feature Selection**

Features are obtained from the input sequence of hand gestures based on the convexity defects found, they are further converted to symbols. Effective and accurate feature vectors play a critical role in K-Curvature method. For selecting good features, the following criteria are considered useful:

(1) Features should be perfectly independent on rotation, translation and scaling.

(2) Features should be easily computable.

(3) Features should be chosen so that they do not replicate each other.

This criterion ensures efficient utilization of information content of the feature vector. The features obtainable from the image sequence of hand gesture are spatial and temporal features. To extract the shape features, we choose the number of defects to describe the hand shape, and to extract the temporal features, we use motion analysis to obtain the non-rigid motion characteristics of the gesture. These features should be invariant to the small hand shape and trajectory variations and it is also tolerant to small different gesture-speed.

The processing of the captured image is shown below:

Video Sequence

Gaussian Filter

Corner Extraction

Boundary Extraction

Noise Removal

Fig 3.4: Flowchart for Image Acquisition and Processing Module

The transmission of signal in order to perform necessary action is shown below:

Image Value

Image value transmission

Receiving Image value

Recognizing Image Value

No

Yes

Value=1 OR 2 OR 3 OR 4?

No Action

Action Performed

Fig 3.5: Flowchart for Signal Transmission

**3.2.2.4.6.3 Fourier Descriptor**

The objects may be described by their features in the frequency domain, rather than those in the spatial domain.

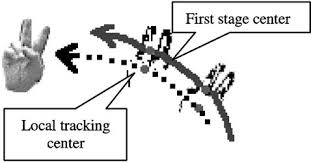


Fig 3.6: Tracking Center of hand

The above figure shows difference between the first stage center and the local-tracking center. The solid line is the trajectory of the first stage center, and dotted line is the trajectory of the second stage center.

The local feature property of the node is represented by its Fourier Descriptors(FD). Assume the hand shape is described byexternal boundary points, then the FD representation may be use for boundary description.

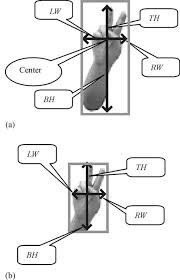


Fig 3.7: FD Representation

**3.2.2.4.5 Sending signal to hardware**

The system includes hardware to control the devices. The hardware includes arduino interfaced with the software IDE to receive signals as and when the gestures are recognized. A bluetooth modem is attached to arduino to receive the signal which is sent by the in-built bluetooth of the laptop. The signal is received as a character and simultaneous operations are performed. Relays are attached with the arduino board to run the devices on the 230V supply. The bluetooth modem is used to make the whole system wireless and for a reliable functioning of the system.

**3.2.2.5 Creating Bluetooth Socket**

**3.2.2.6 Action performed**

The signal is received through bluetooth and the received input thus helps in controlling devices. The devices are attached with the relays. As the signal is received, the corresponding action is performed.

**3.3 Proposed System**

The proposed system presents a special-purpose image processing algorithm that is developed to recognize gestures using convexity defects with high accuracy.

A significant contribution of this result is that it does not require the person making the gestures to use any artificial gloves or markers, hence reducing the cost and reliability.

The proposed system will capture an image using a webcam. The captured image will be converted from RGB to HSV format. Then a threshold filter is applied to remove the non-skin elements.

Gesture normalization is done by the well-known morphological filtering technique, erosion combined with dilation on the extracted image. The output of this stage is a smooth region of the hand figure. Drawing convex hull around the contour and then recognizing the gesture considering the convexity defects found in the extracted region. Now the correct gesture is recognized and the output parameters are passed to arduino to perform expected actions.

The overall working of the Hand Gesture Recognition System has been shown below:

Image Acquisition

Hardware

Image Sequence

Gesture Recognition Module

Feature Extraction

Gesture Classification

Recognized Gesture

Action Based on

Detected Gesture

Fig 3.8: Flowchart for Overall Model

# 

# 4 IMPLEMENTATION AND RESULTS

## 4.1 Implementation Details

Accepting the input: The camera sniffs frames of live video stream at a constant rate

1. Frame Capture: A webcam of 1.3MP or lower is used to capture images of different gestures. The dimension of each image is 640 x 480.

2. Skin Segmentation: The image goes through segmentation and normalization to eliminate the unwanted sections of the image. Only the portion of the image having hand gesture should be selected.

3. Edge detection: The binary image was then converted to an edge image. For edge detection Sobel operator should be used, as the accuracy provided by it is very high. The edge image holds all the significant data that would used for classification

5. Feature Extraction: From the obtained smooth image of hand gesture, we are supposed to identify the correct gesture. For this, K-Cosine algorithm can be used. Gesture is recognized and required parameters are extracted for further processing.

6. Appropriate Action: The parameters required by the arduino are extracted in the previous stage and the signals are sent through Bluetooth. At the receiving end, signals are received by the Bluetooth dongle and specific action is performed.

## 4.2 Results

The system is implemented using C programming language under Microsoft Visual Studio 2013. Our experiments running on a personal computer with Intel i7 processor and 8GB RAM. The video image is captured by the integrated web camera of the laptop with resolution of 640x480.

Since the hand gesture can be in various different types, we define some certain type of gestures to show our experiment result. And we show the results with different number of fingers (0 to 5) under the mentioned condition. Our experimental images captures images in frames, and all the gesture conditions mentioned above are covered.

The recognized gesture is assigned with certain value which is then transmitted to the Bluetooth connected to the electronic device. The value being received is read and the corresponding action is performed.

# 5 TESTING

## 5.1 Test Cases

For implementing gesture recognition, the image was tested using two techniques:

* Background Subtraction
* Skin Segmentation

**5.1.1 Image Testing using Background Subtraction**

Using background subtraction, the first frame is captured and is considered as the background. Then the additional components introduced after some time are considered as the image to be processed. Using this technique, the shadow incase being introduced in the frame is also considered as the significant image to be processed. The contour drawn around such an image is not as good as compared to what is obtained after skin segmentation.

**5.1.2 Image Testing using Skin Segmentation**

Skin segmentation works efficiently for gesture recognition. The webcam selects the skin color being introduced. The introduced hand is then detected based on the HSV values and further preprocessing is done. The image then processed is used to draw contour and find the convexity defects, the number of defects for each gesture is defined and hence the gesture is recognized correctly. This method is highly efficient in ideal conditions and is selected for image processing in the project.

# 6 CONCLUSION AND FURTHER WORK

## 6.1 Conclusion

In this system, we can have an accurate palm and fingertip positions estimation based on a hand contour. We obtained the color image from single web camera, and transform the color space into HSV color space. Our skin color region is defined under HSV color space. A binary image of hand can be obtained according to the skin color definition. Two morphological operations include erosion and dilation are performed. Erosion eliminates the noises while dilation smoothen the boundary. When the usable image is generated, we apply normal contour drawing method to search for the contours in the image. One or more contours can be found, we choose the largest contour as our hand contour. When the hand contour has been chosen, we need to calculate the convex hull of it. We can compare the contour and its convex hull to find all the convexity defects. The contour point inside one convexity defect which has the longest distance to the contour will be the depth point of the convexity defect. By observing the depth points in many different gestures, we notice that the depth points are tending to be around the hand palm. Minimum enclosing circle of all the depth points will be calculated to estimate the palm and the palm center. Also the fingertips can be detected by K-Cosine method. There are some conditions which we might need to modify our estimation. When the amount of depth points are two or less, the estimation of palm could be wrong. We add the highest point in the contour as an extra depth point. Sometimes the depth points beside the wrist will slipped away and enlarge the minimum enclosing circle; which will cause the wrong estimation of palm. We calculate the average position of depth points and mix it with the minimum enclosing circle. This will reduce the effect of the slipped point. The recognized gesture is then given a specific value which is transmitted using Bluetooth. The received value enables the pin on the Arduino microcontroller and the pin set to the particular electronic device is set to HIGH and then that device is turned on. Similarly, when the pin is set LOW that particular device is switched off. Our system has prefect coordination of the hand gesture recognized and the device been controlled.

## 6.2 Further Work

**1) Hand and arm recognition**: Our system so far assumes the input data to be the palm; we can apply hand or arm recognition so that our method can be used in a more flexible condition.

**2) Using stereo camera or depth information**: We can use two cameras from different viewpoint to locate those hand features more accurately. Since the depth camera is a common product nowadays, depth image can be obtained easily. Depth information can be included to know about exact condition of rotation or tilt; and even solve the problem of overlapped hands.

**3) Combine with Augmented Reality**: AR is getting more and more popular recently. Instead of using paper mark, we can use the hand features to locate the virtual objects; and those objects can be interactive to the hand. So the virtual objects won’t be just shown in the video, we can interact with it.

**4) Device can be regulated:** The device can be regulated in terms of speed by using a regulator. This can improvise the utility of the system and can be used efficiently in the home automation system.