

# Social Robot

*A project report submitted in partial fulfillment of the requirements for  
the award of the degree of*

***Bachelor of Technology***

*in*

***Computer Science & Engineering***

Submitted by

Yedhin Kizhakkethara  
Ranjith R.K  
Leo Varghese  
Sandeep Ramesh



**Federal Institute of Science And Technology (FISAT)<sup>®</sup>**  
Angamaly, Ernakulam

*Affiliated to*

**APJ Abdul Kalam Technological University**  
CET Campus, Thiruvananthapuram

**Federal Institute of Science And Technology (FISAT)<sup>®</sup>**  
Mookkannoor(P.O), Angamaly-683577



## CERTIFICATE

This is to certify that report entitled "**Social Robot**" is a bonafide report of the design project (CS341 - Design Project) presented during V<sup>th</sup> semester by **Yedhin Kizhakkethara(FIT16CS125)**, in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (B.Tech) in Computer Science & Engineering.

Staff in Charge

Project in Charge

**Dr. Prasad J C**  
Head of the Department

## **ABSTRACT**

Social Robots soon will be an inevitable part of the distant future in the digital era. A social robot has the ability to see and differentiate the objects in its vision. This feat is achieved by infusing relevant and rather popular computing domains such as Android, Machine learning, Image Processing, Networking etc. Overall, a social robot can imitate certain human features like speaking, movement of head, closing of eyelids and identification of objects in real time. This project aims to create such a social robot which is incorporated with the above mentioned features.

## **ACKNOWLEDGMENT**

The success and final outcome of this project required a lot of guidance and assistance from many people and I am extremely privileged to have got this all along for completing of the project.

I respect and thank APJ Abdul Kalam Technological University (KTU) for providing me an opportunity to do the project work in my college.

I would like to express my deepest appreciation towards **Dr. George Issac**, Principal, Federal Institute of Science and Technology (FISAT), **Dr. Prasad J C**, Head of Department, Computer Science and Engineering, for making the college facilities available at all times.

I am profoundly grateful to **Mr. Mahesh C**, for his expert guidance and continuous encouragement throughout to see that this project reaches its target since its commencement to its completion.

I owe my deepest gratitude to our group advisors **Ms. Resmi R**, **Mr. Jestin Joy** and **Ms. Merin Cherian**, for their guidance and constant supervision as well as for providing necessary information regarding the project.

I am thankful to all other teaching and non-teaching staff for their support in completing the project.

I would also like to thank **Mr. Pankaj Kumar G**, **Ms. Soumya S Raj** and **Mr. Varun P Nair** for helping us out in many aspects of the project and also in setting up the lab for our final presentation.

Last but not the least, I would also like to thank my fellow teammates for their enthusiastic support through coordination and cooperation.

# Contents

<b>List of Figures</b>	<b>v</b>
<b>1 INTRODUCTION</b>	<b>1</b>
1.1 Overview . . . . .	1
1.2 Objective . . . . .	1
1.3 Scope Of Project . . . . .	1
1.4 Robotics . . . . .	2
1.5 Characteristics . . . . .	3
1.6 Application . . . . .	3
<b>2 RELATED WORK</b>	<b>8</b>
<b>3 DESIGN and IMPLEMENTATION</b>	<b>11</b>
3.1 Design . . . . .	11
3.1.1 UML Deployment Diagram . . . . .	11
3.1.2 Data Flow Diagram - Level 0 . . . . .	12
3.1.3 Data Flow Diagram - Level 1 . . . . .	13
3.1.4 3D-Designed Model . . . . .	14
3.2 Support Vector Machine(SVM) . . . . .	15
3.3 System Requirements . . . . .	15
3.3.1 Hardware Requirements . . . . .	15
3.3.2 Software Requirements . . . . .	16
3.4 Implementation . . . . .	16
3.4.1 Dataset Collection . . . . .	17
3.4.2 Android App Development . . . . .	17
3.4.3 Server Creation . . . . .	17
3.4.4 Image Classification . . . . .	17
3.4.5 Hardware Implementation . . . . .	17
3.4.6 Hardware Control . . . . .	18
3.4.7 Libraries . . . . .	19
<b>4 TESTING</b>	<b>20</b>
4.1 Feature Extraction Methods . . . . .	20
4.2 Arduino Controlling Methods . . . . .	20
4.3 Asynchronous Networking . . . . .	21
<b>5 RESULTS</b>	<b>22</b>
<b>6 CONTRIBUTIONS</b>	<b>24</b>
<b>7 CONCLUSION</b>	<b>25</b>

<b>Appendices</b>	<b>27</b>
<b>A Feature Extraction Using HOG</b>	<b>28</b>
<b>B Support Vector Machines(SVM)</b>	<b>29</b>
<b>C Retrofit POST Request API</b>	<b>30</b>

# List of Figures

1.1	Line Follower Robot . . . . .	2
1.2	NASA Mars Rover . . . . .	4
1.3	Robots in Military Field . . . . .	5
1.4	Robots in Household Applications . . . . .	5
1.5	Robots in Industrial Applications . . . . .	6
1.6	Robot in Health Services . . . . .	7
2.1	Kismet . . . . .	8
2.2	Joe Robot . . . . .	9
2.3	Furby Robot . . . . .	9
3.1	Front view of the robot . . . . .	14
3.2	Movements of the robot . . . . .	14
5.1	Our Social Robot . . . . .	22
5.2	Rotation of head . . . . .	23

# List of Tables

4.1	Testing of Feature Extraction Methods . . . . .	20
4.2	Testing of Arduino Controlling Methods . . . . .	21
4.3	Testing of Asynchronous Networking Methods . . . . .	21

# **Chapter 1**

## **INTRODUCTION**

### **1.1 Overview**

A social robot is an autonomous robot that interacts and communicates with humans or other autonomous physical agents by following social behaviors and rules attached to its role.

### **1.2 Objective**

The objective of the project is to make a social robot which can enact certain human features like identification of objects seen through its vision, speaking out about the objects which it can see, movement of the head, and closing of eyelids, by making use of popular domains such as android, machine learning, image processing and networking.

### **1.3 Scope Of Project**

The social consequences of robotics depend to a significant degree on how robots are employed by humans, and to another compelling degree on how robotics evolves from a technical point of view.[1] That is why it could be instructive for engineers interested in cooperating with sociologists to get acquainted with the problems of social work and other social services, and for sociologists interested in the social dimensions of robotics to have a closer look at technical aspects of new generation robots.[2]

Interactive robotic toys like Hasbro's Baby Alive My Real Babies; household companions such as Sony's AIBO dog, Jetta's robotic dinosaur Pleo, and Aldebaran's NAO next generation robot; therapeutic pets like the Paro baby seal; and the Massachusetts Institute of Technology (MIT) robots Kismet, Cog, and Leonardo.[3] The use of humanoid robots to comfort and entertain lonely older persons has already triggered an ethical debate.[5]

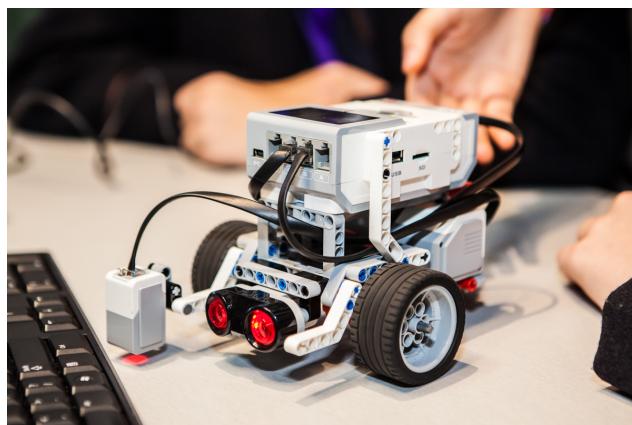
The relevance of social robots should not be underestimated. In technologically advanced societies, a process of robotization of social work is already underway.[8]

For instance, robots are increasingly used in the care of the elderly. This is a consequence of two other processes occurring simultaneously: on the one hand, we have an aging population with a resulting increase in demand for care personnel; on the other hand, technological developments have created conditions to deal with this problem in innovative ways.

## 1.4 Robotics

Robotics can be described as the current pinnacle of technical development. Robotics is a confluence science using the continuing advancements of mechanical engineering, material science, sensor fabrication, manufacturing techniques, and advanced algorithms. The study and practice of robotics will expose a dabbler or professional to hundreds of different avenues of study. For some, the romanticism of robotics brings forth an almost magical curiosity of the world leading to creation of amazing machines. A journey of a lifetime awaits in robotics.[6, 7]

Robotics can be defined as the science or study of the technology primarily associated with the design, fabrication, theory, and application of robots. While other fields contribute the mathematics, the techniques, and the components, robotics creates the magical end product. The practical applications of robots drive development of robotics and drive advancements in other sciences in turn.



**Figure 1.1:** Line Follower Robot

The promise of robotics is easy to describe but hard for the mind to grasp. Robots hold the promise of moving and transforming materials with the same plan and ease as a computer program transforms data. Today, robots mine minerals, assemble semi-processed materials into automobile components, and assemble those components into automobiles. On the immediate horizon are self-driving cars, robotics to handle household chores, and assemble specialized machines on demand.[9, 2, 3] It is not unreasonable to imagine robots that are given some task, such as reclaim desert into photo-voltaic cells and arable land, and left to make their own way. Then the promise of robotics exceeds the minds grasp. In summary, robotics is the field related to science and technology primarily related to robotics. It stands tall by standing the accomplishments of many other fields of study.

## 1.5 Characteristics

**Sensing:** First of all your robot would have to be able to sense its surroundings. It would do this in ways that are not dissimilar to the way that you sense your surroundings.

**Movement:** A robot needs to be able to move around its environment. Whether rolling on wheels, walking on legs or propelling by thrusters a robot needs to be able to move. To count as a robot either the whole robot moves, like the Sojourner or just parts of the robot moves, like the Canada Arm.

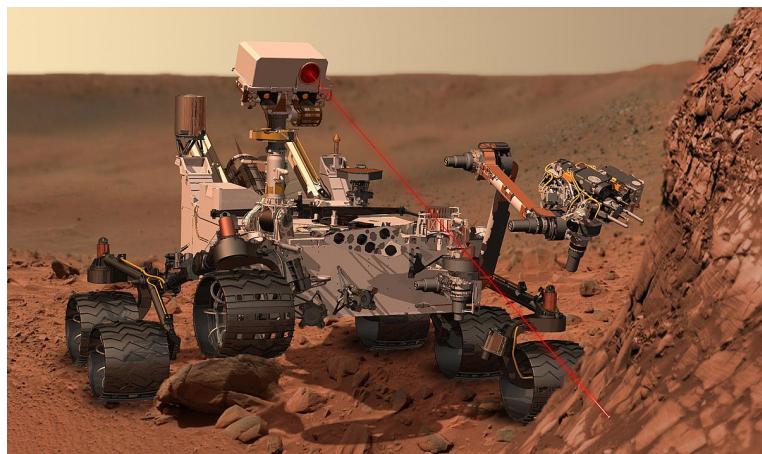
**Energy:** A robot needs to be able to power itself. A robot might be solar powered, electrically powered, battery powered. The way your robot gets its energy will depend on what your robot needs to do.

**Intelligence:** A robot needs some kind of "smarts." This is where programming enters the pictures. A programmer is the person who gives the robot its 'smarts.' The robot will have to have some way to receive the program so that it knows what it is to do.

## 1.6 Application

Robotics is the engineering science and technology which involves the conception, design, operation and manufacture of robots. Electronics, mechanics and software are brought together by robotic. Robots are used for jobs that are dirty, dull and dangerous. Today robotics have many different application areas. Some of those are:

**Outer Space Applications:** Robots are playing a very important role for outer space exploration. The robotic unmanned spacecraft is used as the key of exploring the stars, planets...etc. Sojourner Mars Explorer Robot The most famous robots used in the outer space applications are the Mars rovers of NASA. In 1997 The Pathfinder Mission landed on Mars. Its robotic rover Sojourner, rolled down a ramp and onto Martian soil in early July. It continued to broadcast data from the Martian surface until September. Sojourner performed semi-autonomous operations on the surface of Mars as part of the Mars Pathfinder mission; equipped with an obstacle avoidance program. Sojourner was capable of planning and navigating routes to study the surface of the planet. Sojourner's ability to navigate with little data about its environment and nearby surroundings allowed the robot to react to unplanned events and objects.[9]



**Figure 1.2:** NASA Mars Rover

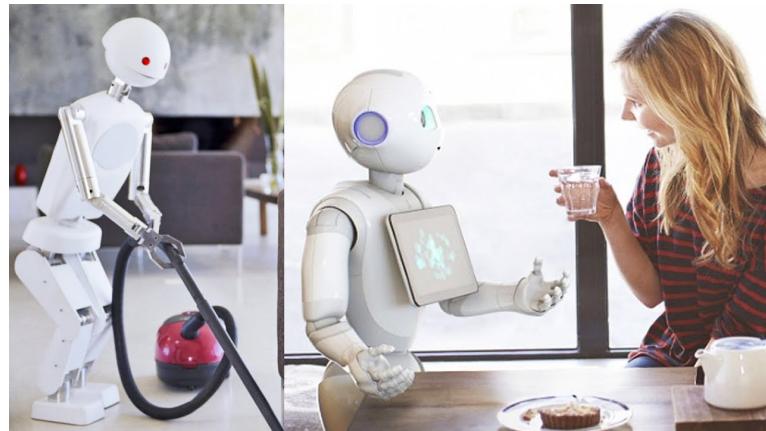
Spirit and Opportunity Mars Rovers After Sojourner's mission NASA sent twin robots Spirit and Opportunity to the Red Planet on 10, June and 23, July 2003. Spirit and Opportunity landed on Mars on 4, January and 25, January 2004. Spirit and Opportunity are solar powered robots with six wheels included their own motors. Both of the Mars Rovers are 1,5 m high, 2,3 m wide and 1,6 m long and weighing 180 kg. Spirit and Opportunity have many science instruments in order to perform their missions on Mars. They have a robot arm, that contains a Mössbauer spectrometer to investigate the mineralogy of the rocks and soils on Mars, an Alpha particle X-ray spectrometer for analysis of elements found in rocks and soils, a rock abrasion tool used to expose the fresh material for examination, a microscopic image and magnets to collect magnetic particles. Phoenix Mars Explorer Robot The twin Mars Rovers have a panoramic camera used for examinations of the texture, color, mineralogy, and structure of the local terrain, a miniature thermal emission spectrometer for identification promising rocks and soils which is useful to determine the formation processes of them. There is also a navigation camera on both Mars rovers in order to take view with a higher field but lower resolution for driving and navigation.[10, 11] Phoenix Mars Rover was sent to the Red Planet on 4, August 2007 and landed on 25, May 2008. The mission of the Phoenix was to investigate the existence of water and life supporting conditions on Mars.

**Military Applications:** In today's modern army robotics is an important factor which is researched and developed day by day. Already remarkable success has been achieved with unmanned aerial vehicles like the Predator drone, which are capable of taking surveillance photographs, and even accurately launching missiles at ground targets, without a pilot. There are many advantages in robotic technology in warfare however, as outlined by Major Kenneth Rose of the US Army's Training and Doctrine Command: "Machines don't get tired. They don't close their eyes. They don't hide under trees when it rains and they don't talk to their buddies...A human's attention to detail on guard duty drops dramatically in the first 30 minutes ... Machines know no fear."



**Figure 1.3:** Robots in Military Field

**Intelligent Home Applications:** We can monitor home security, environmental conditions and energy usage with intelligent robotic home systems. Door and windows can be opened automatically and appliances such as lighting and air conditioning can be pre-programmed to activate. This assists occupants irrespective of their state of mobility.



**Figure 1.4:** Robots in Household Applications

**Industry:** From the beginning of the industrial revolution robotics and automation becomes the most important part of manufacturing. Robotic arms which are able to perform multiple tasks such as welding, cutting, lifting, sorting and bending are used in fabrics.



**Figure 1.5:** Robots in Industrial Applications

The most commonly used configurations of the industrial robots are:

- \* Articulated Robots: An articulated robot is one which uses rotary joints to access its work space. Articulated robots can range from simple two-jointed structures to systems with 10 or more interacting joints. The six-axis, articulated robot is the most versatile industrial robot which allows for a high level of freedom.
- \* Cylindrical Coordinate Robots: These robots have three degrees of freedom and they moves linearly only along the Y and Z axes with a cylindrical work envelope.
- \* Scara Robots: It stands for Selective Compliant Assembly Robot Arm or Selective Compliant Articulated Robot Arm. Scara robots usually have four axes as any X-Y-Z coordinate within their work envelope and a fourth axis of motion which is the wrist rotate (Theta-Z).
- \* Spherical Coordinate Robots: The spherical arm, also known as polar coordinate robot arm, has one sliding motion and two rotational, around the vertical post and around a shoulder joint.
- \* Cartesian Coordinate Robots: Rectangular arms are sometimes called "Cartesian" because the arms axes can be described by using the X, Y, and Z coordinate system. It is claimed that the Cartesian design will produce the most accurate movements.
- \*Delta Robots: A Delta robot consists of three arms connected to universal joints at the base. The key design feature is the use of parallelograms in the arms, which maintains the orientation of the end effector. The Delta robot has popular usage in picking and packaging in factories.[2]

Health Service: Under development is a robotic suit that will enable nurses to lift patients without damaging their backs. Scientists in Japan have developed a power-assisted suit which will give nurses the extra muscle they need to lift their patients - and avoid back injuries.[3]

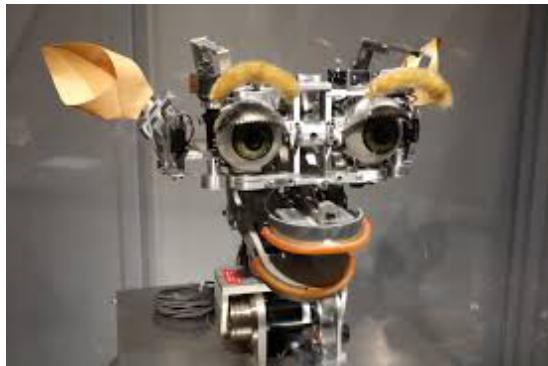


**Figure 1.6:** Robot in Health Services

# Chapter 2

## RELATED WORK

- Kismet
  - It's a robot made at MIT as an experiment in effective computing.
  - Kismet simulates emotions through various facial expressions, vocalizations and movement.



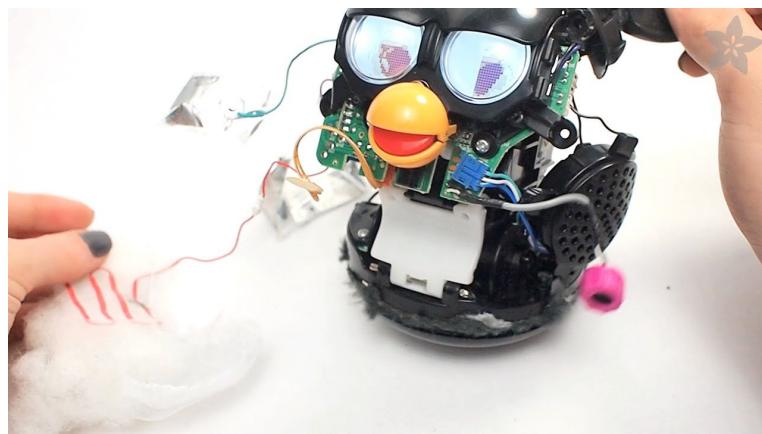
**Figure 2.1:** Kismet

- Joe Robot
  - It's a robot made in research of Anthropomorphism.
  - It develops intelligence during the meaningful social interactions between AI and people.



**Figure 2.2:** Joe Robot

- Furby
  - It's a electronic robotic toy created by Tiger Electronics.
  - Furby initially speaks in furbish and then later on learns the language that the owner speaks by itself.



**Figure 2.3:** Furby Robot

Furby is an American electronic robotic toy released in 1998 by Tiger Electronics. It resembles a hamster or owl-like creature and went through a period of being a "must-have" toy following its holiday season launch, with continual sales until 2000. Over 40 million Furbies were sold during the three years of its original production, with 1.8 million sold in 1998, and 14 million in 1999. Its speaking capabilities were translated into 24 languages.

Furbies were the first successful attempt to produce and sell a domestically-aimed robot. A newly purchased Furby starts out speaking entirely "Furbish", the unique language that all Furbies use, but is programmed to start using English words and phrases in place of Furbish over time. This process is intended to resemble

the process of learning English.[1] The updated Emoto-Tronic Furby, with voice-recognition and more complex facial movements, was sold by Hasbro between 2005–2007. They released another updated Furby with LCD eyes and a mobile app for the holiday season in 2012.[5]

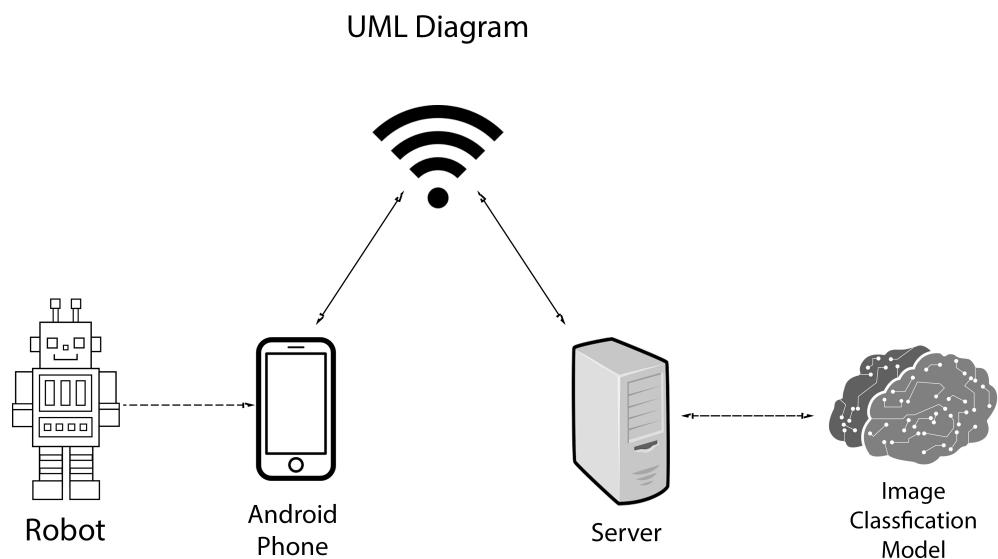
# Chapter 3

## DESIGN and IMPLEMENTATION

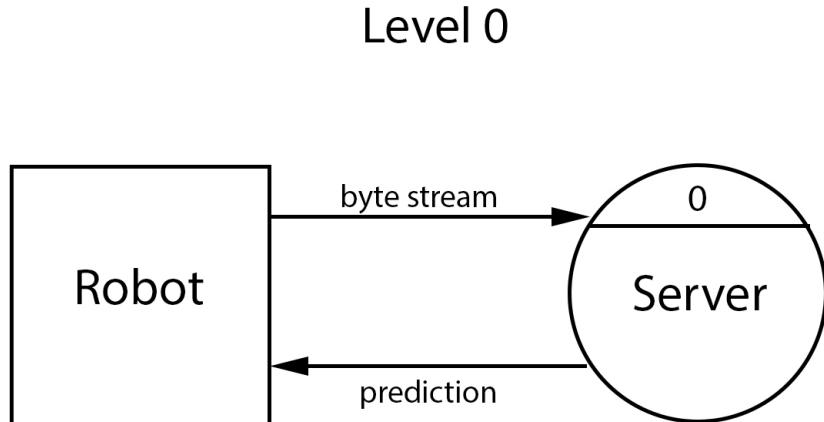
### 3.1 Design

The proposed work is to create a social robot which can identify objects in its visibility. The physical body of the robot will have human like movement capabilities, like movement of head, closing of eyelids and speaking. A web camera will act as the eyes of the robot. The live feed from the web camera will be directed to an android app and send via WiFi to the server, where classification will take place. An image classification model trained using Support Vector Machine(SVM) will be used in the process of classification. The prediction will be send back to the android app which would then act as the mouth of the robot and speak out what it sees.

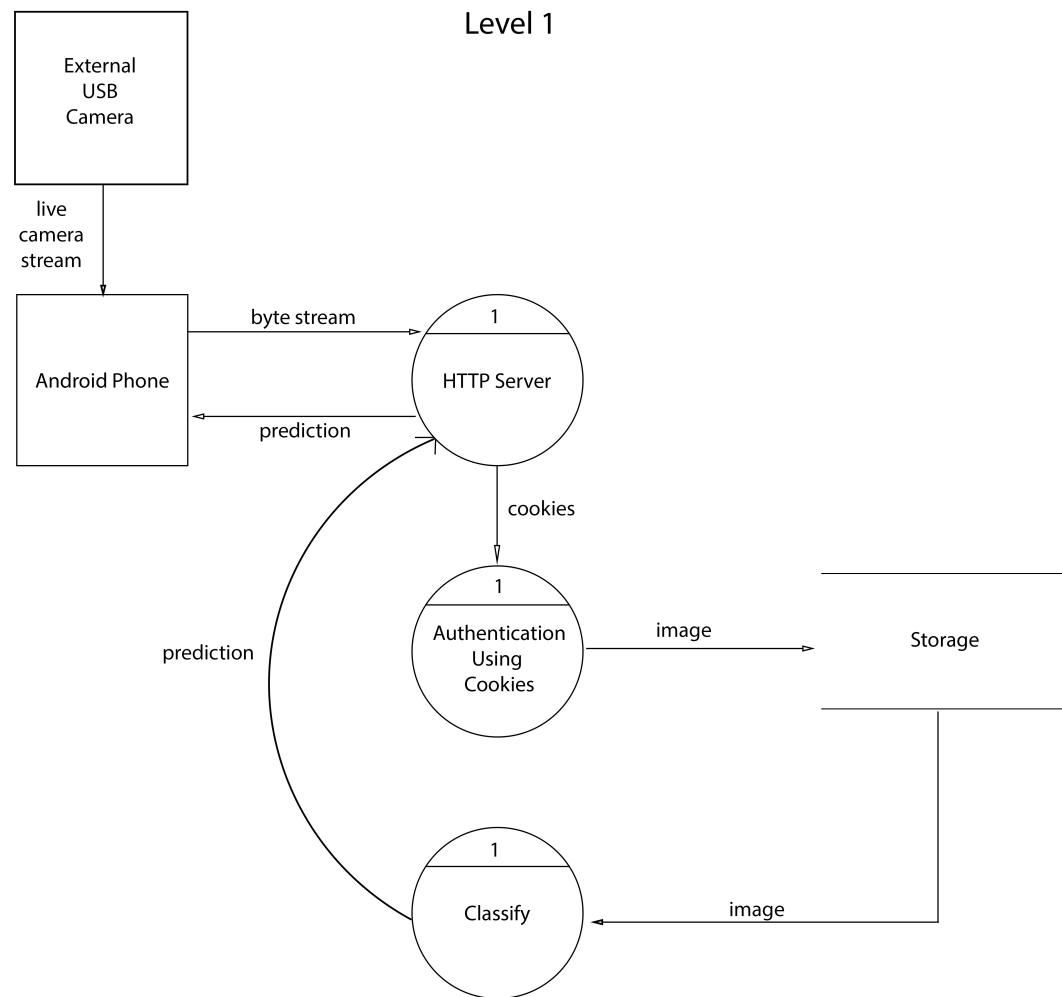
#### 3.1.1 UML Deployment Diagram



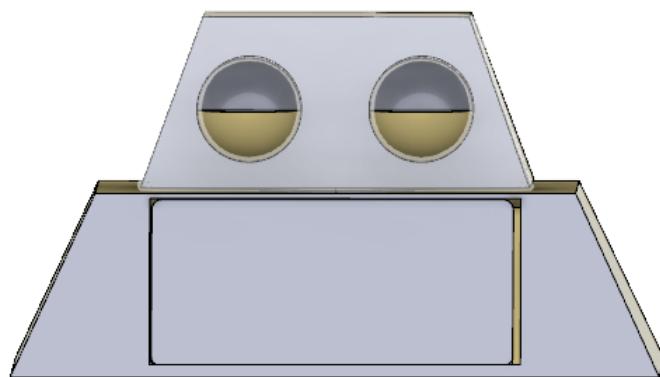
### 3.1.2 Data Flow Diagram - Level 0



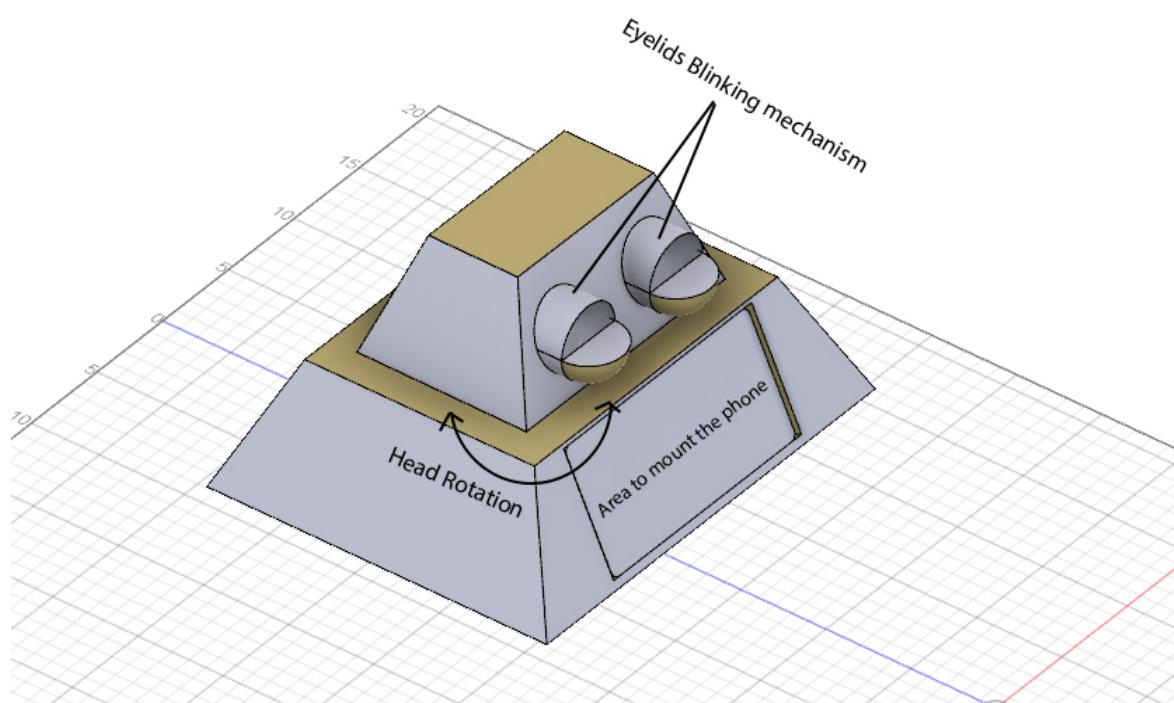
### 3.1.3 Data Flow Diagram - Level 1



### 3.1.4 3D-Designed Model



**Figure 3.1:** Front view of the robot



**Figure 3.2:** Movements of the robot

## 3.2 Support Vector Machine(SVM)

In machine learning, support vector machines[12] (SVMs, also support vector networks) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier (although methods such as Platt scaling exist to use SVM in a probabilistic classification setting). An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

In addition to performing linear classification, SVMs can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces.

The main reasons why we used SVM over other popular image classification techniques is because:

- Training a model using SVM doesn't require that much computational power compared to deep learning and ANN models.
- HOG + SVM combination provides the best possible accuracy and performance among the supervised learning techniques.
- SVMs deliver a unique solution, since the optimality problem is convex. This is an advantage compared to Neural Networks, which have multiple solutions associated with local minima and for this reason may not be robust over different samples.

## 3.3 System Requirements

### 3.3.1 Hardware Requirements

1. Arduino
2. Mega servo motor
3. Mini servo motor
4. USB Camera
5. Smart Phone

### 3.3.2 Software Requirements

1. Image Classification Modules
  - (a) opencv
  - (b) sklearn
  - (c) numpy
  - (d) scipy
  - (e) scikit-learn
  - (f) h5py
2. Server Modules
  - (a) python-tornado
3. Dataset Collection Modules
  - (a) scrapy
4. Android Modules
  - (a) Retrofit 2
  - (b) UVC Camera
  - (c) GOTEV(speech)
5. Arduino Control Modules
  - (a) PyFirmata
  - (b) Firmata Protocol code for APL

## 3.4 Implementation

A simple flow of control can be described as follows:

The usb camera provides vision of the robot and this data is fetched by the android phone which then sends the data stream to the server, where classification takes place and the prediction is send back to the android app itself, and then the prediction is spoken out.

On a deeper level the implementation part can be divided into several stages.

- Dataset Collection
- Android App Development
- Server Creation
- Creation Of Image Classification Model
- Hardware Side Modelling and Creation
- Hardware Control

### 3.4.1 Dataset Collection

The datasets were collected by creating a web crawler, that crawled sites like amazon, flipkart and other sites which had stock photos in large numbers. The crawler was created using the Scrapy library. The datasets were organized automatically into their respective directories(where directory name was the class label) using a python script. In total each of the class labels consisted of nearly 3500 images.

### 3.4.2 Android App Development

The android phone acts as a middle-ware between the robot and the server. The UVCCamera library was used to retrieve the live stream from the usb camera. Retrofit, a type safe REST Client library, was used in communicating with the server and most importantly for sending and receiving data from the server. The usb camera stream was converted to byte stream and it was sent to the server along with a unique id, which can act as the cookie, to the http server. The server sends back the predictions based on the data and it's received in the android app itself. This prediction is then spoken out using the Speech library.

### 3.4.3 Server Creation

An http server was created using the python-tornado library. The server can handle multiple users at a time. Most importantly, the server makes use of the image classification model to predict on the received data and send back the prediction to the client(which is the android app). The trained image classification model stored in the server will detect objects within a few milli seconds.

### 3.4.4 Image Classification

The Support Vector Machine(SVM) algorithm was used to train a image classification model. This classification technique was applied in training by making use of the inbuilt functions from the Sklearn library in python. Feature extraction using the concept of Histogram of Oriented Gradients(HOG) was implemented using opencv library. The pre-processing of the data(that is scaling and normalizing) was done using sklearn itself. For choosing the best hyper-parameters, GridSearch Tuning was also implemented. The encoded labels and all were stored in h5py files, which acted as containers, datasets and groups and allowed for easier debugging as well classification. Finally the trained image classification model is stored to the file system for later usage.

### 3.4.5 Hardware Implementation

The body of the social robot was made using Multi-Wood, which is a white sheet made up of 'U' Pvc polyester resin, as it's more easily moldable and handy. A usb camera was mount inside the head of the robot, and it acted as the eyes of the robot. The external eye and eyelids were 3D-printed for precision. Rotation

of head and eyelids were done using servo motors. A total of 3 servo motors, 1 for rotating the head and 2 for blinking of eyelids were used.

### 3.4.6 Hardware Control

Arduino Uno was made to control the overall movement of the servo motors. The control signals for arduino were sent from python, by making use of the pyserial and pyfirmata libraries. The rotation of head, and closing of eyes were synchronized using python itself. The important part, which was making the robot face towards a "person/human" in its vision, was also done by synchronizing the movements based on the predictions made.

### 3.4.7 Libraries

OpenCV : OpenCV 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.[7]

Numpy : Numpy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.[7]

Sklearn and Scikit-learn : Scikit-learn (formerly scikits.learn) is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to inter-operate with the Python numerical and scientific libraries NumPy and SciPy.[10]

Retrofit : Retrofit is a REST Client for Java and Android. It makes it relatively easy to retrieve and upload JSON (or other structured data) via a REST based webservice. In Retrofit we can configure which converter is used for the data serialization. Typically for JSON you use GSon, but we can add custom converters to process XML or other protocols. Retrofit uses the OkHttp library for HTTP requests.[11]

UVCCamera : It's a native script to get access to UVC web camera on non-rooted Android devices and do further processing with the data stream.[11]

# Chapter 4

## TESTING

Testing was done at several stages of the project. It was mainly used in choosing the best feature extraction method, in choosing a language for controlling arduino and for a server that can handle multiple users at a time.

### 4.1 Feature Extraction Methods

Test Method	Expected Outcome	Outcome
HOG	Counts occurrences of gradient orientation in localized portions of an image	Accurate feature descriptors were obtained
SURF and SIFT	Keypoint detection and description	Key points were detected, but the algorithms were not used since they are not free to use(patended)
HU-MOMENTS	Outline or “silhouette” of an object in an image	The feature descriptor was not enough for an accurate prediction

Table 4.1: Testing of Feature Extraction Methods

### 4.2 Arduino Controlling Methods

Test Method	Expected Outcome	Outcome
Arduino Programming Language(APL)	Control each and every movement of the servo motors and overall synchronization	Not as expected. Dynamic control of the servos is not possible
Python(Standalone)	Dynamic Control and ease of use	Was more dynamic, but since the servos are controlled through a serial port, sending multiple control signals simultaneously proved erroneous

Python + APL	Accurate synchronization and control of servos	The firmata protocol allowed parallel executions and use of external libraries from python, combined with a start-up code from APL
--------------	--	--

**Table 4.2:** Testing of Arduino Controlling Methods

### 4.3 Asynchronous Networking

Test Method	Expected Outcome	Outcome
Python-Flask	Ease of use	Limited in handling multiple requests at a time
Python-Tornado	Asynchronous Networking and speed	Was able to handle upto 10 users at a time simultaneously, without compromise in speed.

**Table 4.3:** Testing of Asynchronous Networking Methods

# **Chapter 5**

## **RESULTS**

The output of the project is a social robot which can identify the objects seen in its vision. The robot, in its physical structure, has a 180 degree view made possible by its head movements.

The robot performs a specific function of object classification. The live feed it retrieves from the usb camera attached to its eyes are fed to an android application on the phone. The app send the images to the server where it undergoes a number of processes as specified in the explanation of the project. Once the image classification is done, the result is displayed on the phone screen.



**Figure 5.1:** Our Social Robot



**Figure 5.2:** Rotation of head

It has numerous civilian and military applications. But, on a undergrad level, what we aim to achieve is to bring high level of accuracy on live classification of objects and to incorporate with a physical model like a robot. What we can do with that, is according to its applications.

# **Chapter 6**

## **CONTRIBUTIONS**

As I was the team leader, I had a definitive role to play in the overall completion of the project. I undertook many important responsibilities like getting the tasks done on time by using all of the available resources and co-ordinating with my teammates, for dividing and assigning tasks to teammates, for acting as the conduit between the college management, the teaching staff and my team, and most importantly played the role of a core developer in all stages and domains of the project.

The contributions made by each of the teammates were very much valuable to the completion of the project. Key contributions and roles made by each of the teammates is as follows :

- Ranjith R.K : Designed the 3D model, collected and organized the dataset, processed images and extracted features from it.
- Leo Varghese : Core developer in Android section of the project, implemented the ability to speak for the robot, sets up the vision of the robot, helped in crafting the hardware.
- Sandeep Ramesh : Implemented an image classification model using SVM and also helped in the hardware side.

## **Chapter 7**

### **CONCLUSION**

A social robot is an autonomous robot that interacts and communicates with humans or other autonomous physical agents by following social behaviors and rules attached to its role. This project successfully created a social robot, incorporated with human features such as, movement of head, closing of eyelids and speaking, and can identify and speak out about the objects seen through its vision, by making use of useful and popular computing techniques such as Android, Machine learning, Image processing and Servers.

## Bibliography

- [1] Blackford, R. 2012. Robots and reality: A reply to Robert Sparrow. *Ethics and Information Technology* 14(1): 41–51.
- [2] Campa, Riccardo. 2015. Humans and automata. A social study of robotics. Frankfurt am Main: Peter Lang.
- [3] Daerden, F., and D. Lefeber. 2000. Pneumatic artificial muscles: Actuators for robotics and automation. Vrije Universiteit Brussel.
- [4] Darling, Kate. 2012. Extending legal rights to social robots. Paper presented at We Robot Conference. University of Miami, April 23.
- [5] Desai, J.P., G. Dudek, O. Khatib, and V. Kumar, eds. 2013. Experimental robotics. Heidelberg: Springer.
- [6] Flandorfer, Priska. 2012. Population ageing and socially assistive robots for elderly persons: The importance of sociodemographic factors for user acceptance. *International Journal of Population Research Article ID* 829835.
- [7] Floreano, D., and C. Mattiussi. 2008. Bio-inspired artificial intelligence: Theories, methods, and technologies. Cambridge, MA: MIT Press.
- [8] Ge Shuzhi, S., H. Li, J.-J. Cabibihan, Y.K. Tan, eds. 2010. Social robotics. Second international conference. Proceedings. Heidelberg: Springer.
- [9] Application of social robot:<http://www.robotiksistem.com/robotics-applications.html>
- [10] Free software machine learning library for the Python programming language:<https://scikit-learn.org/>
- [11] Android SDK tools and API documentation:<https://developer.android.com/studio/>
- [12] Cortes, Corinna; Vapnik, Vladimir N. (1995). "Support-vector networks". *Machine Learning*. 20 (3): 273–297. doi:10.1007/BF00994018.

# Appendices

## Appendix A

### Feature Extraction Using HOG

```
import cv2
import imutils
import storage as st
from trained_models import model

def histogram_of_oriented_gradients(image):
    winSize = (100, 100)
    blockSize = (10, 10)
    blockStride = (5, 5)
    cellSize = (10, 10)
    nbins = 9
    derivAperture = 1
    winSigma = -1.
    histogramNormType = 0
    L2HysThreshold = 0.2
    gammaCorrection = 1
    nlevels = 64
    signedGradients = True
    hog = cv2.HOGDescriptor( winSize , blockSize ,
                           blockStride , cellSize ,
                           nbins , derivAperture ,
                           winSigma , histogramNormType ,
                           L2HysThreshold , gammaCorrection ,
                           nlevels , signedGradients )
    return hog . compute( image )

image = cv2.imread( st . __UPLOADS__ + fname )
image = imutils.rotate_bound( image , 90 )
#resize the image
image = cv2.resize( image , ( 100 , 100 ) )
# convert to grayscale
image = cv2.cvtColor( image , cv2.COLOR_BGR2GRAY )
# extract features
gf = histogram_of_oriented_gradients( image )
# make prediction
prediction = model.predict( gf . reshape( 1 , -1 ))[0]
print( prediction )
```

## Appendix B

### Support Vector Machines(SVM)

```
from sklearn import svm
# we have , X (predictor) and Y (target) for training data set
# and x-test (predictor) of test dataset
# Create SVM classification object
model = svm.svc(kernel='linear', c=1, gamma=1)
# there is various option associated with it , like changing kernel ,
# gamma and C value .
# Train the model using the training sets and check score
model.fit(X, y)
model.score(X, y)
#Predict Output
predicted= model.predict(x_test )
```

## Appendix C

### Retrofit POST Request API

```
package com.ikigai.android.imageupload.apis;

import okhttp3.MultipartBody;
import okhttp3.RequestBody;
import okhttp3.ResponseBody;
import retrofit2.Call;
import retrofit2.http.Multipart;
import retrofit2.http.POST;
import retrofit2.http.Part;

public interface RetrofitInterface {
    @Multipart
    @POST("/upload")
    Call<ResponseBody> uploadImage(@Part MultipartBody.Part image);
}
```