King Fahd University of Petroleum and Minerals
Computer Engineering Department

Senior Design Project (COE485)
Semester 142

2/28/2015

Autonomous Person-Following Robot

Design Document

Team Members:

- Adnan AlSinan 201015940
- Abdulrahman Bataweel 200927570
- Husam Habarah 200971590

Advisors:

- Dr. Muhamed Mudawar
- Dr. Yahya Osais

Table of Contents

Table of Figures

Table of Tables

1 Introduction

Every day; we see shopping carts in the supermarket. Customers usually push the carts with both hands, and so, if the customer had only one hand or had to carry a child, then pushing shopping carts becomes a real burden.

Thus, in order to improve the situation and assist those with disability, we plan to develop a robot system that follows the user. Enabling disabled people to enjoy shopping with the need for human assistance (pick and drop on the robot).

1.1 How the project deals with the issue

People live in their lives in many modes. There is the young and adults mode where they are able to help themselves and help others. But there is certain stages in life that people needs help from others to carry their substances. It is embarrassing to ask help from anyone that might either happy accept, or might sadly refuse. This project aims to help those people to avoid asking others for help. A Robot will help in carrying there substances and follow them in a convenient way without disturbing their attention in shopping.

1.2 Project Impact on society locally and globally

1.2.1 Positive impact

Personal assistant robot is one of the promising areas where robotics could be put in practice. Nowadays, approaching "aging society" could make use of robots by helping people in their everyday life. To keep a better quality of life of elderly people, a robot could move with people like a partner and carry items without disturbing anyone by avoiding obstacles. Moreover, it is possible to use this robot to monitor and keep an eye at kids preventing them from hurting themselves or messing with electronic appliances. Beyond that, this robot could be used in different areas such as military, rescuing missions and medical applications.

2 Problem Statement

Autonomous robots have the capability of gaining information about the environment. They work without the need for human intervention for a long period of time. They can also adapt to changes in their surrounding environment.

In order to assist elderly or disabled people in supermarkets, our goal is to design and build a robot that is capable of following them, and carry weight for them, to substitute for pushing a shopping cart.

3 Background

3.1 Existing Solutions, Products & Research

3.1.1 Nippon Institute of Technology, Japan.

They were able to develop a robot shopping cart, the robot successfully recognized its user and followed him through the use of Laser Range Sensor (LRS). The sensor could detect objects in the distance of 4m and in the azimuth of 240 degrees.

However, this solution suffered a major flaw, when a third person cut across between the user and the robot shopping cart, the cart ended up following the third person. Minor flaws included, interference with the following procedure from obstacles such as shopping shelves and display tables.

Figure 1: Robot Shopping Cart

3.1.2 R&D Center, Toshiba Corporation, Japan.

ApriAttenda is a more sophisticated robot that uses cameras, LRF and ultrasonic sensors. ApriAttenda adopts a stereo vision system and additionally a LRF is mounted on it to enhance the performance of person following motion. The designed tracking system uses highly accurate measurement information by combining vision and LRF data according to the congestion level of movement space.

An experiment of person following was done using ApriAttenda using the sensor fusion method. And from this experiment, it was observed and confirmed that the robot can follow a person smoothly who moves quickly and randomly.

Figure 2: ApriAttenda Robot

4 Requirements and Specifications

4.1 Functional user requirements.

- The robot should be able to follow a specific person
- It should be able to recognize the person from different directions.
- It should be able to recognize obstacles such as aisles, counters, and people and avoid them.

4.2 Non-functional user requirements.

- The robot should be able to carry at least 1 kg of weight.
- The robot can follow a subject within 2-5 M.
- The robot should not exceed a power consumption of 2100AMH.
- Response time should be within 2 seconds

4.3 Technical Specification.

- Identify subject to follow by camera.
- Following subject identified.
- Robot can carry up to 1 kg.
- 180 degree of view supported by servo motor move horizontally.
- 5 photo per second to catch subject movement.
- Movement speed is 6 km/h to Follow-up human at maximum speed of 6 km/h.
- Keep distance of <2 meters to the followed person.
- Length adjustment of camera between 1.5 -2 meter.

5 System Design

5.1 Solution Concept

So the stated problem is, to create a robot that is able to follow its user and carry weight for them. That by itself, can be divided into two sub-problems, first is person detection, and second is person following. So, in all three solution concepts mentioned below, we are going to show how each one generally solves the problem.

5.1.1 General approach of solving the stated problem.

Description:

The approach we are proposing here with ultimatly provides the best results. consist of using camera to detect the user of the robot, and distance sensors to measure the distance between the user and the robot.

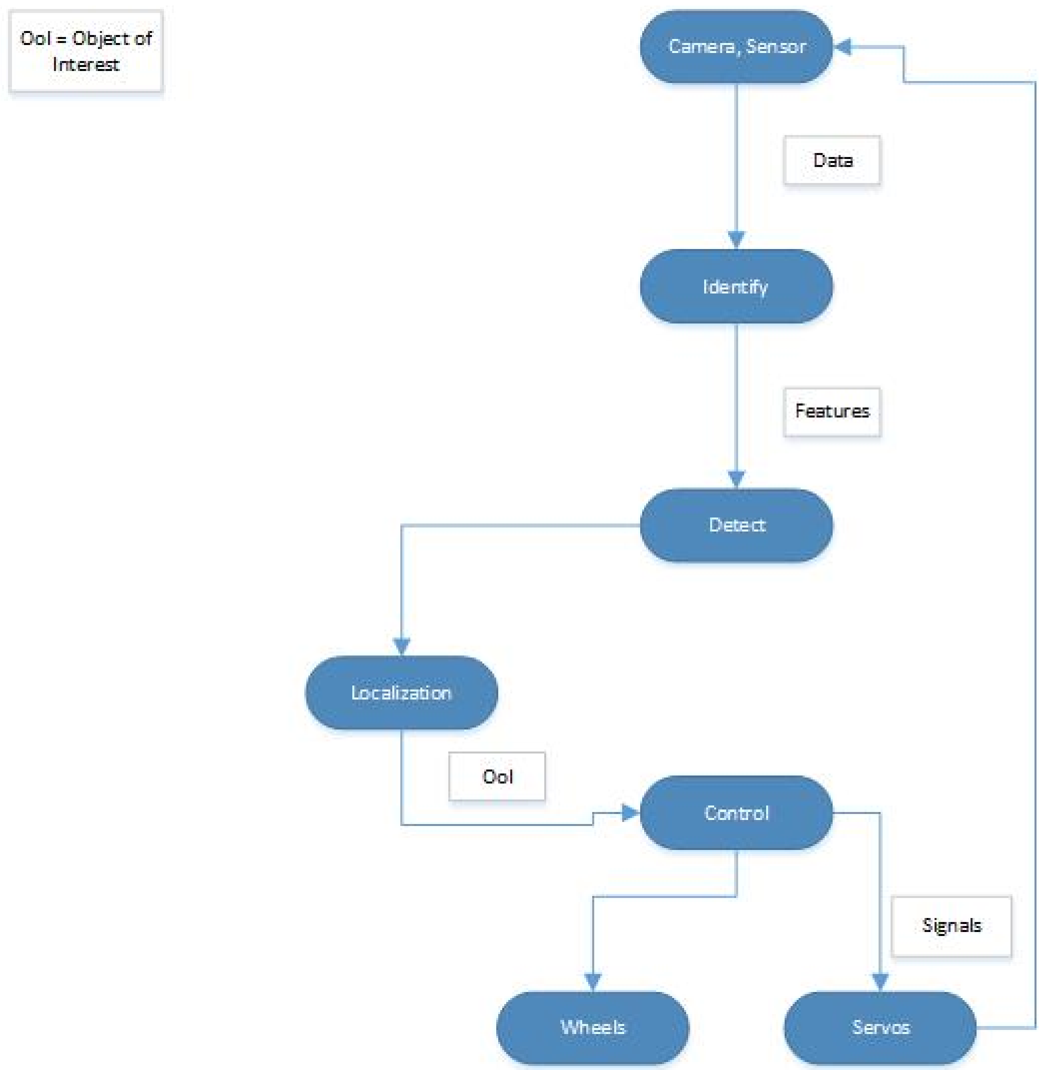


Figure 3: General Approach

5.1.2 Description of used/developed algorithms

- Image processing algorithm

Histogram of Oriented Gradients (HOG) is feature descriptors used in computer vision and image processing for the purpose of object detection. The technique counts occurrences of gradient orientation in localized portions of an image. This method is similar to that of edge orientation histograms, scale-invariant feature transform descriptors, and shape contexts, but differs in that it is computed on a dense grid of uniformly spaced cells and uses overlapping local contrast normalization for improved accuracy.

The people detector object detects people in an input image using HOG and a trained Support Vector Machine (SVM) classifier. The object detects unclouded people in an upright position.

- Follow and motion algorithm

This algorithm should provide control over the Robot motors. To drive the robot forward or backward, or turn to any direction with provided angel. For simplicity, we will start with one parameter for each function: move time (move for t seconds). Work should be done to determine the accurate measurement for the time given over distance covered.

5.1.3 Alternative approaches and algorithms.

Approach 2:

The alternative approach would use RFID technology in order to detect the person.

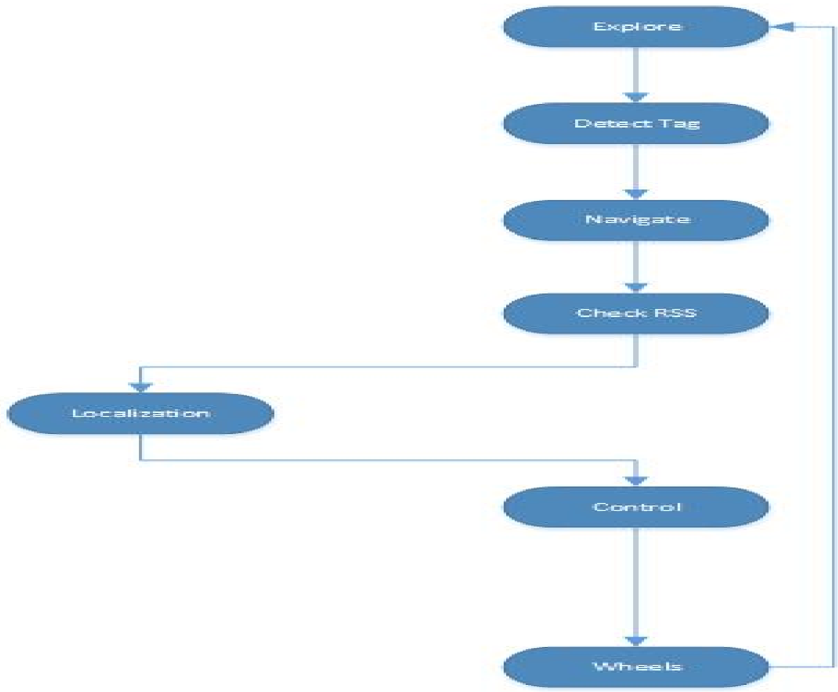


Figure 4: Alternative Approach # 2

Approach 3:

The third alternative approach is just using an infrared sensor (IR) to detect the person and follow him. However by just using a sensor, this approach is far beyond from being autonomous or smart. This approach rely directly on infrared sensor to detect the person and send commands to move the cart. However as mentioned previously, this approach fails when a third person cuts in between the robot and the user.

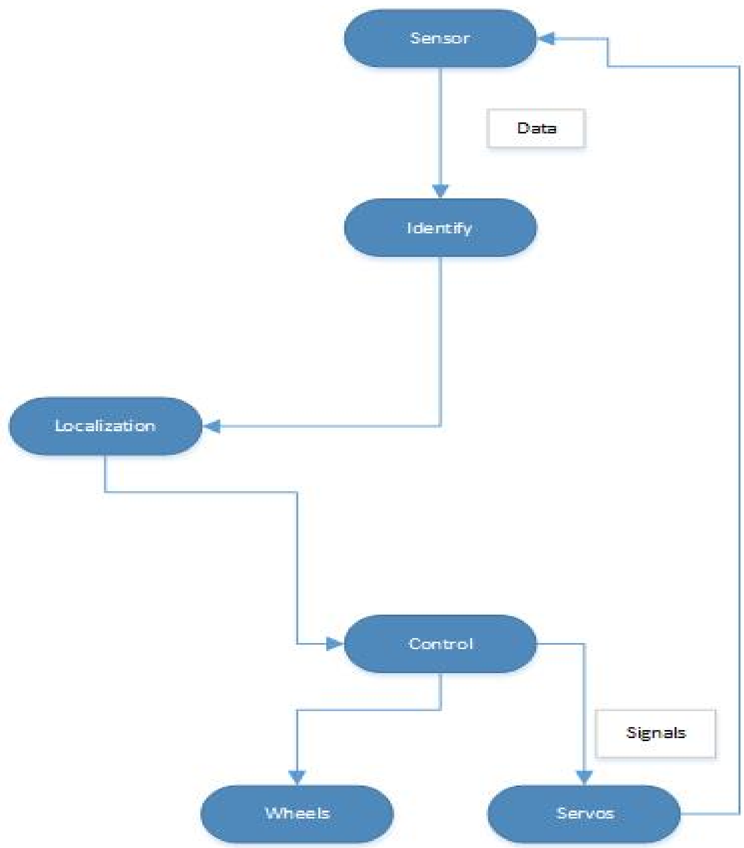


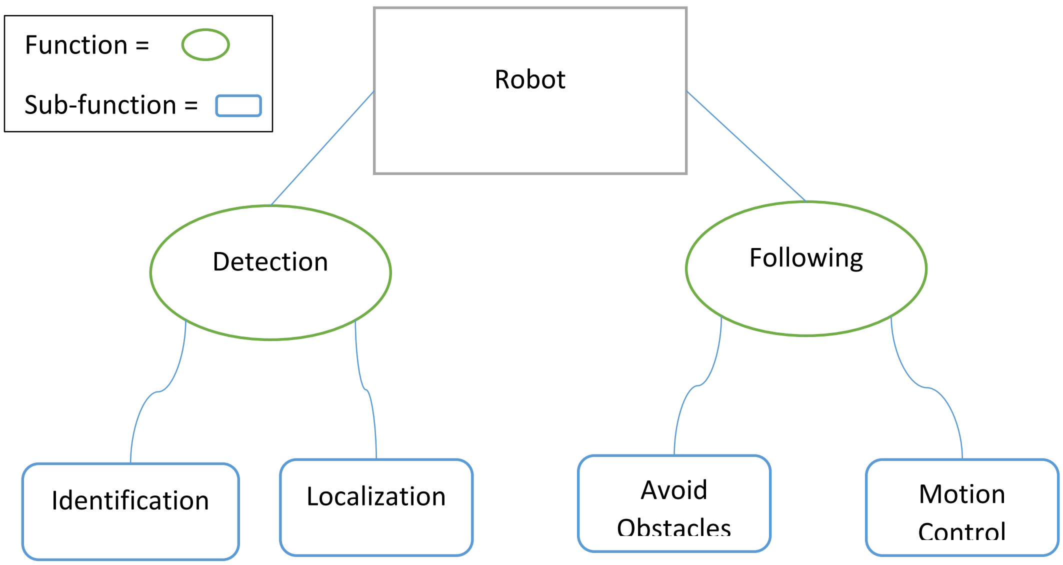
Figure 5: Alternative Approach # 3

5.1.4 Comparison, and selection criteria

	Criteria					
Approach	Range	Accuracy	Invasiveness	Processing Time	Deployment	Price
RFID, with active tags	4 meters – 50 meters	Very accurate	Need to wear a tag or a device	Depends on the type of the tag	Two sides, on robot and on user	$ 500 - $ 2000
IR	1 meter – 3 meters	Not accurate, a third person might cut between.	No	Negligible	On the robot	$15 - $ 30
Camera and sensors	No specific range; dependent on the sensors	90%. [1]	No need to wear a tag nor to carry a device	the average processing per frame is about 51.76 ms.[2]	Everything will be on the robot.	Camera $300 - $400 Sensor $6 - $10

Table 1: Approach Comparison Table

5.1.5 Sub-function identification



5.2 Architecture

5.2.1 System architecture and components

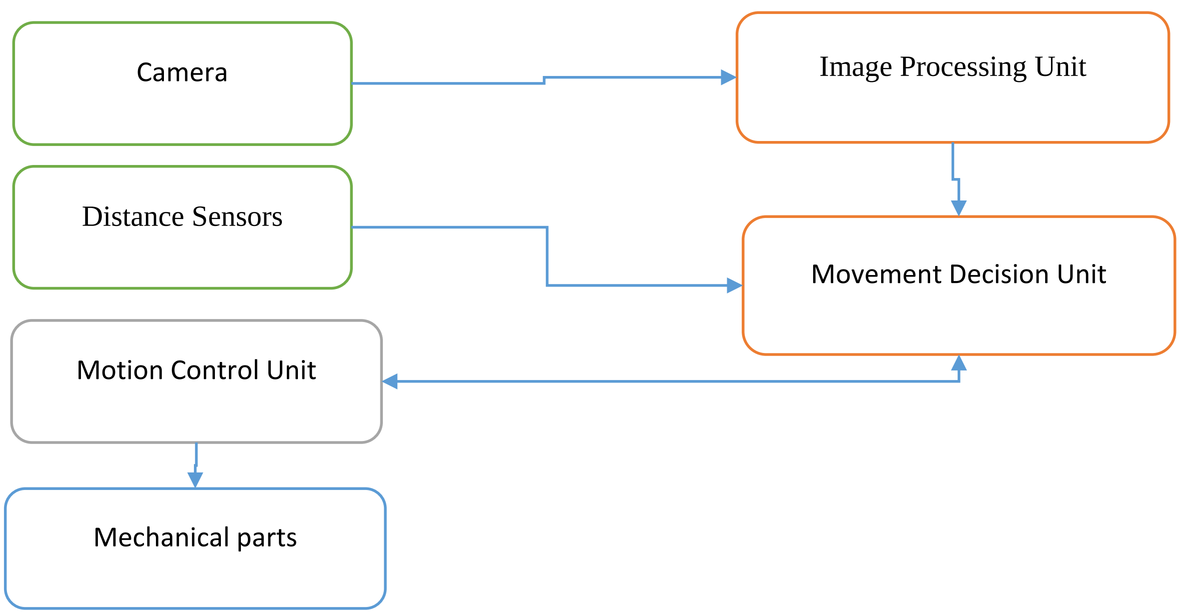


Figure 6: System Architecture

5.2.2 Alternative architectures

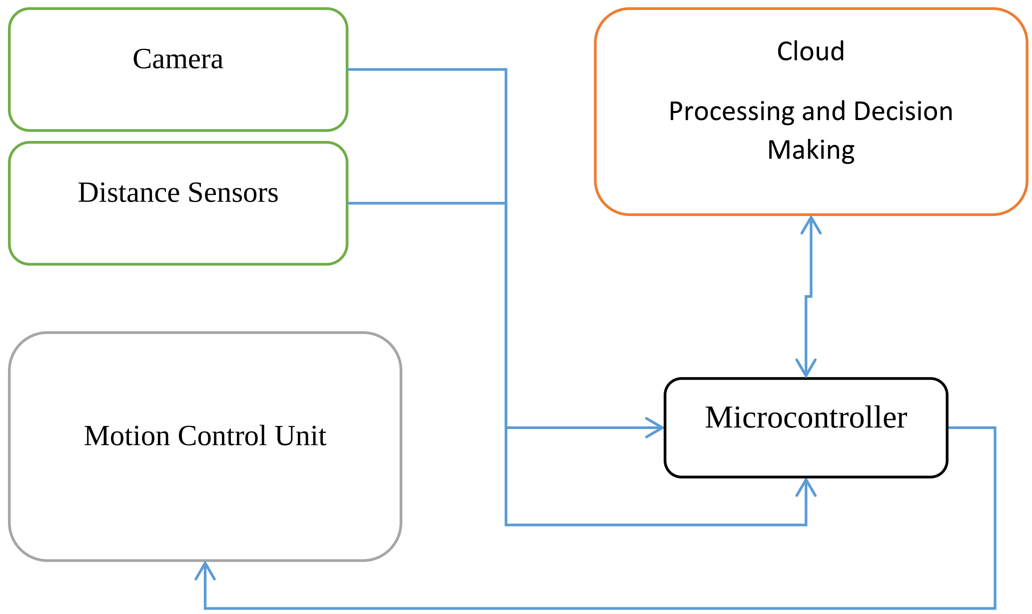


Figure 7: Alternative Architecture

5.2.3 Comparison, and selection criteria

Based on the following criteria, we have chosen the first architecture.

ARCHITECTURE/CRITERIA	PROCESSING TIME	POWER CONSUMPTION & BATTERY LIFE	LATENCY	ADDITIONAL COST
ARCHITECTURE 1 (WITH AN IMAGE PROCESSING UNIT)	5 frames/sec, 1 frame = 50 kb 250kb x 60 = 21MB/min	Would last for 8 hours	Negligible	None
ARCHITECTURE 2 (WITH CLOUD)	5 frames/sec, 1 frame = 50 kb 250kb x 60 = 21 MB/min (and that's just to upload the image to the cloud, add to it the processing time of the image itself then the download time)	Since, it requires the device to be connected to the internet all the time. Would last for about 3 hours.	In best case milliseconds; dependent in the Cellular network	Cellular network ; 1 GB/month 79 riyals

Table 2: Architectures Comparison

5.2.4 Hardware vs software components

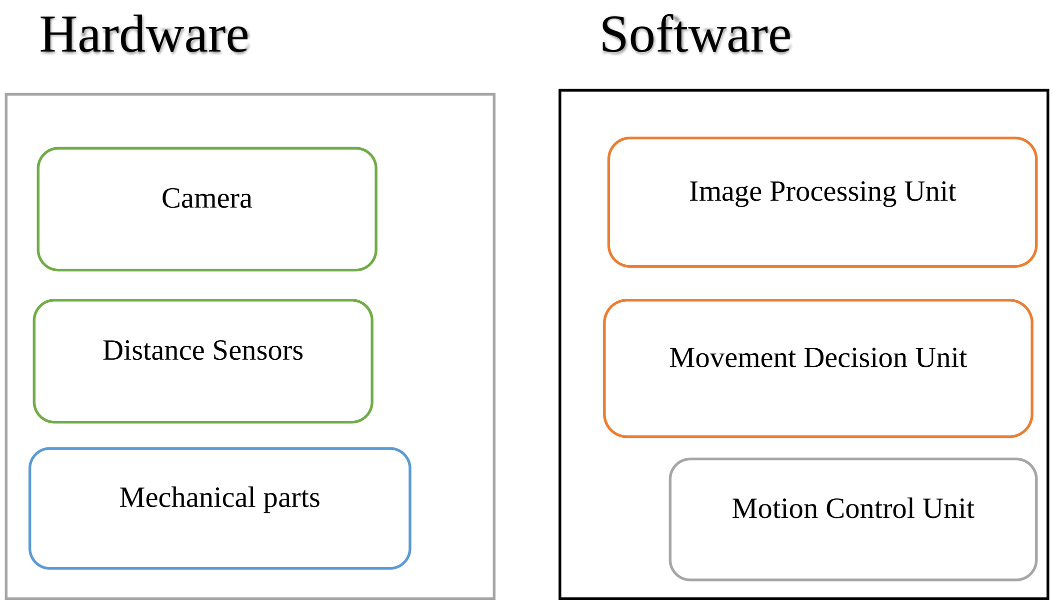


Figure 8: Hardware vs Software

5.2.5 Functions of each component

- **Camera**: Image input
- **Distance Sensors**: To measure the distance between the user and the robot, for the purpose of knowing whether he moved or not.
- **Motion Control Unit**:
  - Provide signals to move the robot based on the output from the movement decision unit
  - Control the servos
- **Mechanical Parts**:
  - Like wheels, servos.
- **Image Processing Unit**:
  - Receives the image from the camera and extract features of the object of interest in it.
- **Movement Decision Unit**
  - Decide the movement of the robot (whether it goes right, left, forward or backward) based on the sensors and image processing results.
  - Keeps the robot line of sight synchronized.

5.3 Component Design

5.3.1 Motion Control Unit

For the **motion control unit** we're going to use the T'Rex Robot/Motor Controller

The T'Rex controller from DAGU is an Arduino compatible robot controller designed to power and control servos and brushed motors. This controller combines an Arduino development board with a dual FET H-bridge motor driver. The heart of the board is an ATMega328P AVR microcontroller

The T'REX controller is rated for a **maximum voltage of 30V** and can handle currents in excess of **40A per motor**. A set of screw terminals are provided for **the battery connection** as well as for **connecting motor outputs.** The Dual H bridges are rated for stall currents of 40A per motor and average currents of 18A per motor.

Features:

- 6V -30V operation with built in solid state power switch
- Programmable with the Arduino IDE (ATMega328P, 5V @ 16MHz)
- Dual FET "H" bridge rated 18A with self-resetting PTC fuses
- Electronic braking and current monitoring for each motor
- 3-axis accelerometer provides angle and impact detection
- Auto-detects RC, Bluetooth, or I^2C control
- Voltage translation on I^2C interface
- 6x Servo Outputs

5.3.2 Distance Sensors

For the distance sensors, we're going to use ultrasonic sensors

Features:

For the servos, we're going to use Hitec HS-422 (Standard Size)
Features:

- Voltage: 4.8-6.0 Volts
- Torque: 45.82/56.93 oz-in. (4.8/6.0V)
- Speed: 0.21/0.16 sec/60° (4.8/6.0V)
- Direction: Clockwise/ Pulse Traveling 1500-1900usec
- Rotation: 180°
- Dual Oilite Bushing
- Nylon Gears
- 3-Pole Ferrite Motor
- C1 Standard Spline

5.3.3 Bluetooth Module

For the Bluetooth, we're going to use DAGU Bluetooth module, to communicate between the android and the T'Rex microcontroller.

Features:

- Input voltage 3.3v ~ 6v, module with 3V LDO voltage regulator
- Support AT command set the baud rate, name, pair password
- Search and pairing automatically when the power up

- Power, link status indicator
- 4pin 2.54 Pitch status indicator

5.3.4 Image Processing and Movement Decision Units

For the Image processing and movement decision units, we had four options; an android device, an iPad/iPhone device, Beaglebone black, raspberry pi 2 rev b.

So based on the comparison, we have decided to go with the android device option. As it has almost everything built-in in it. And it is easy to develop in it, as it uses java.

	Criteria								
Option	OS	CPU	Storage	RAM	Camera	Bluetooth	Connectivity	Cost	Development platform
Beaglebone black	Linux	Dual-core1.3GHz	4GB eMMC, and micro SD	512MB DDR3	Not built in	Not Built in	Ethernet	SR 190	C, C++, Python, Perl, Ruby, Java, or even a shell script
Rasberry PI 2 B	Can run linux/windows	Quad-core900MHz	Micro SD	1GB	Camera interface	Not built in	Ethernet	SR 150	C, C++, Python, Perl, Ruby, Java.
Android device	Android	Quad-core 2.3GHz Dual-core 2.3GHz	16 GB	2GB	8 MP	v4.1, A2DP, apt-X	Wifi, GSM	SR 1450	Free to develop. Uses Java as a programming language for development
IPhone 5s	iOS 8.1.3	Dual-core 1.3GHZ	16 GB	1GB	8 MP	v4.0, A2DP	Wifi, gsm	SR 2225	Restrict development, must pay for developer ID. Languages are restricted to Swift and objective C.

Table 3: Image Processing and Movement Decision Units options comparison

	Criteria	
Option	Frame/second	Cost
Android built-in Camera	1080 @30 fps	Included with the android device
IPhone built-in Camera	1080 @30 fps	Included with the iPhone device
Logitech c920	1080 @ 5(at jpeg format)-30(at h264 format) fps	SR 400

Logitech c310	720 @ 15 fps	SR 300
Logitech c270	720 @ 15 fps	SR 200

Table 4: Cameras options comparison

5.4 System Integration

5.4.1 Interfaces

For system Integration, **Decision Making Unit** should receive input from the **Image processing unit** in the form of *object location on 180 degree of camera view*, and from **distance sensor** in form of *distance between robot and object*. By complaining those two input in algorithm it should provide a decision whether the robot will move or not, and in which angel to move and how much distance should be covered to keep the specified distance between robot and object.

One more input is needed to determine if there is an **obstacle** in the direction specified or not, this should be processed by the Decision making Unit by using provided inputs from **distance sensor** and **Image processing** output.

To interface this with the Motor (or Motion) Unit, output should be send to Moving Control Unit in form of distance and angle to be executed by functions in the MCU.

A Distance sensor integrated to T'Rex to provide input to the Follow-Person algorithm.

5.4.2 Sequence Diagram

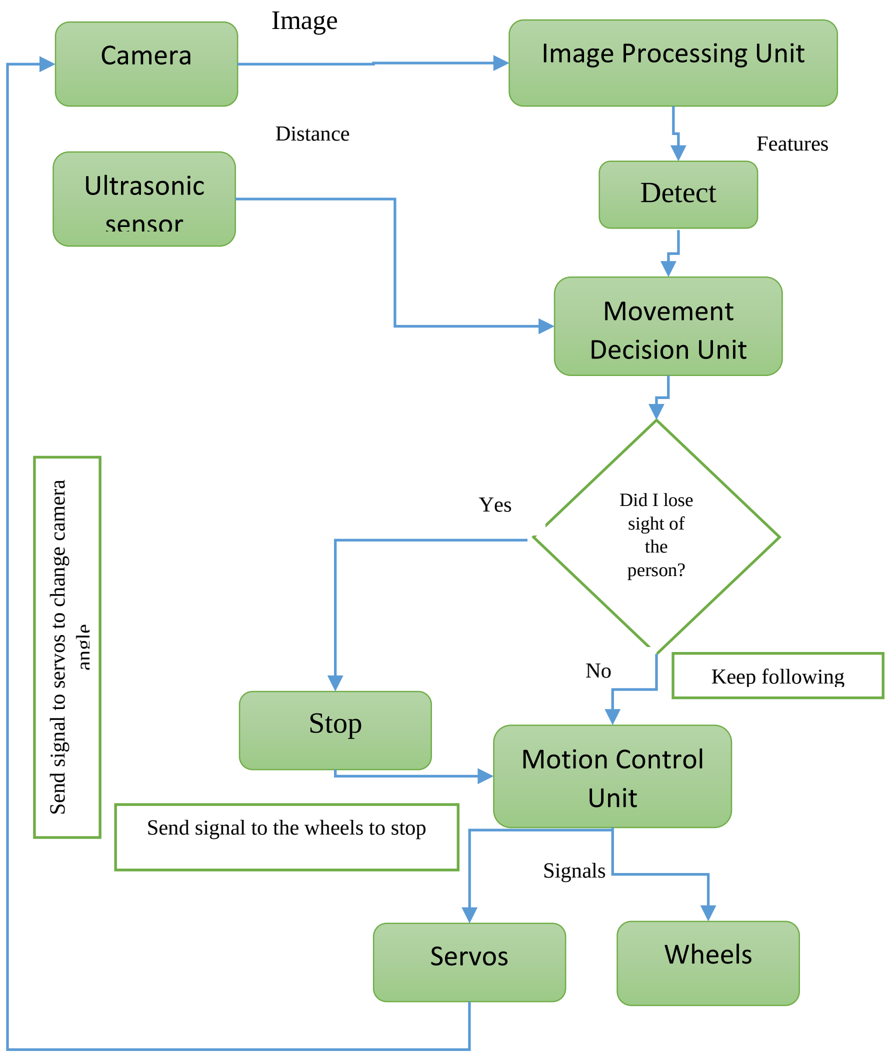


Figure 9: Component sequence diagram

5.5 Progress

The following tables will show the project tasks along with their description and duration, as well as the team accomplished tasks so far.

Tasks	**Owner**	**Duration**		**Status**
		Start Date	End Date	
1. *Project Plan*	Husam	1/25/15	2/2/15	Completed
2. *Collecting the requirement*	Husam	2/1/15	2/7/15	Completed
2.1. *Collect the requirement*	Husam	2/1/15	2/1/15	Completed
2.2. *Review with the advisor*	Husam	2/2/15	2/2/15	Completed
2.3. *Researching the idea*	Adnan	2/3/15	2/3/15	Completed
2.4. *Finalizing the requirements*	Abdulrahman	2/4/15	2/7/15	Completed
2.5. *Finishing the specifications*	Abdulrahman	2/8/15	2/8/15	Completed
3. *Designing*	Abdulrahman	2/9/15	2/28/15	Waiting
3.1. *Researching solutions of similar problems*	Adnan	2/9/15	2/9/15	Completed
3.2. *Discuss and compare the different solution concepts to solve the problem*	Abdulrahman	2/10/15	2/11/15	Completed
3.3. *Evaluate the different solution concepts*	Husam	2/12/15	2/14/15	Completed
3.4. *Discuss the different approaches for image processing and person detection*	Adnan	2/15/15	2/18/15	Completed
3.5. *Evaluating the different approaches*	Abdulrahman	2/19/15	2/20/15	Completed
3.6. *Discuss and compare the different system architectures*	Adnan	2/21/15	2/22/15	Completed
3.7. *Evaluate the different architectures*	Husam	2/23/15	2/24/15	Completed
3.8. *Evaluate the different component options*	Husam	2/24/15	2/25/15	Completed
3.9. *Finalize the hardware components*	Adnan	2/22/15	2/25/15	Completed

Table 5: Tasks progress

3.10. Check that all the hardware components work	Abdulrahman	2/22/15	2/26/15	Delayed – Missing Bluetooth module
3.11. Design Document	Husam	2/15/15	2/28/15	Completed
3.12. Proof of concept	Adnan	2/29/15	3/3/15	In progress
4. Implementation	Adnan	3/3/15	5/8/15	In progress
4.1. program the motion control unit	Abdulrahman	3/3/15	4/23/15	In progress
4.2. program the Image processing Unit	Adnan	3/3/15	4/27/15	In progress
4.3. Connecting the motors and the board	Husam	3/3/15	4/30/15	In progress
4.4. Coding the person-following algorithm	Adnan	3/3/15	5/3/15	In progress
5. Testing	Husam	5/3/15	-	Waiting

6 References

1. Takafumi, K., Onozato, T., Tamura, H., Katayama, S., & Kambayashi, Y. (n.d.). Design of a Control System for Robot Shopping Carts. Retrieved February 1, 2015, from http://leo.nit.ac.jp/~tamura/pdf/1780.pdf
2. Sonoura, T., Yoshimi, T., Nishiyama, M., Nakamoto, H., Tokura, S., & Matsuhira, N. (n.d.). Person Following Robot with Vision-based and Sensor Fusion Tracking Algorithm. Retrieved February 4, 2015, from http://cdn.intechopen.com/pdfs-wm/5205.pdf
3. Tarokh, M., & Merloti, P. (2010). Vision-Based Robotic Person Following under Light Variations and Difficult Walking Maneuvers. *Journal of Field Robotics, 27*(4). Retrieved February 2, 2015, from http://onlinelibrary.wiley.com/
4. Cameron, S. (n.d.). *A comparison of keypoint descriptors in the context of pedestrian detection: Freak vs. surf vs. brisk*. Informally published manuscript, Standford University CS Department, . Retrieved from http://cs229.stanford.edu/proj2012/Schaeffer-ComparisonOfKeypointDescriptorsInTheContextOfPedestrianDetection.pdf
5. Tian, Q., Zhou, B., Wei, Y., & Fei, W. (2013). Human Detection using HOG Features of Head and Shoulder Based on Depth Map. *Journal of Software, 8*(9). Retrieved February 10, 2015, from http://ojs.academypublisher.com/index.php/jsw/article/view/9783
6.