EC535 Project Report

Smart Trash Can

By:

Abhishek Gaur

Anubhav Gupta

Ritika Chatterjee

I. Abstract

With the relentless innovation in the home automation and smart systems, a lot of emphasis is being paid to automating devices like televisions, air conditioning systems, security systems etc. But we realized one thing that hasn't received a lot of attention in this process, that is, our trash can. A smart home doesn't look smart enough if it's dirty. Hence, we made a "Smart Trash Can" as a proof of concept to what could the trash cans look like in these smart homes. This project designs and implements an autonomous moving trash can that tracks a person willing to throw trash using Microsoft's Kinect Camera, actuate the motors of the moving base and walks up to the person to grab the trash. It can be built very easily using less complex hardware and we see a lot of potential in the concept being applied to not just trash cans but other things as well like laundry bags, wardrobe etc. And hence, we propose this project as a proof of concept to what could be a really useful product for smart homes.

II. Introduction

Smart Trash Can, as the name suggests is a trash can that tracks a person willing to throw the trash and moves autonomously towards him/her and grabs the trash. The sensor utilized to track the person was Microsoft's Xbox Kinect Camera. Image processing was done using OpenCV running on a Windows machine. Location of the trash can was determined using color segmentation technique. The base of the trash can moved using simple DC motors which received the distance and direction data on a Gumstix board using Bluetooth.

a) Motivation

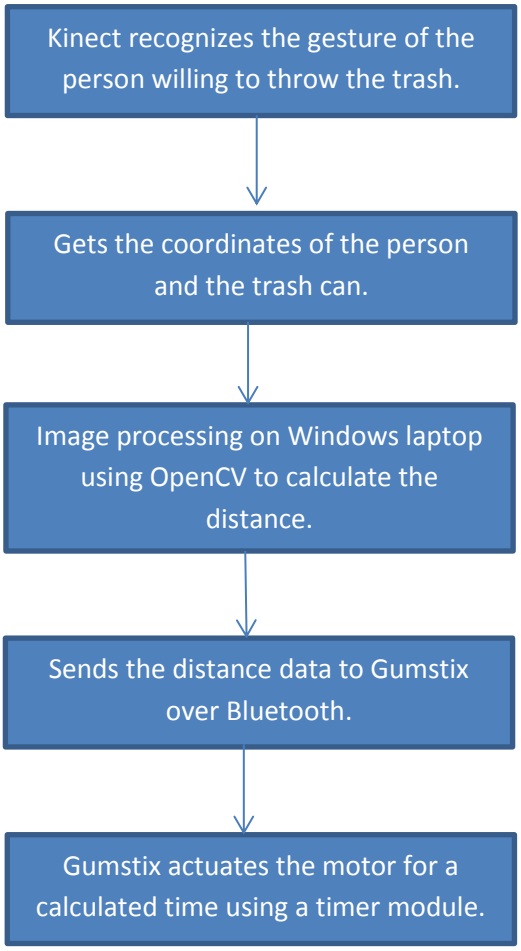
With the amount of innovation going on in the smart homes and smart systems, we wanted to build something that could potentially be a part of our future homes and make our lives easier. After digging through a lot of smart systems available in market, we realized that a trash can is something which is an irreplaceable part of any home and is not smart enough, yet. Hence we decided to do something related to that. After a discussion within ourselves and our peers at school, we realized that finding a trash can and walking up to it every time you want to throw some trash is something each one of us would like to avoid.

The problem can be very easily be dealt with if all this effort that has to be done by us can be done by the trash can itself. Hence, we decided to build a trash can that is capable of doing this, at a very basic level, thereby providing a prototype of what the trash cans could do in the near future. The problem at hand is critical since keeping our surroundings clean is our civic responsibility, but at the same time is a tedious task at times given that every other daily chore we perform have been made so much more easier owing to some great products in the market and advent of Internet of Things. The basic idea was to reduce human effort and at the same time make sure that the task that needs to be done gets done in an efficient manner.

b) Results:

The trash can we built successfully managed to do the task of tracking the person every single time he/she wants to throw the trash and thereby traveled towards him/her and grabbed the trash. The success rate of the operation was nearly 95% and the trash can operated only the predefined gesture and didn't respond to any stray gesture a person might perform unknowingly. Also, the person was not required to hold the pose was while waiting for the trash can to arrive and just needed to signal it once. The response at times was a little delayed owing to the complexity attached with Kinect Camera, less processing power of the Gumstix board and range on Bluetooth module. That said, the trash can performed the aforementioned task successfully and was pretty swift in moving towards the person.

Design Flow


Kinect recognizes the gesture of the person willing to throw the trash.
Gets the coordinates of the person and the trash can.
Image processing on Windows laptop using OpenCV to calculate the distance.
Sends the distance data to Gumstix over Bluetooth.
Gumstix actuates the motor for a calculated time using a timer module.

III. Project Details:

We designed and implemented the following modules:

a) **"Gesture"** module in C++ to utilize the image and infrared data from Kinect to track the person willing to throw the trash and get the position coordinates of the person and trash can to find the distance between them.
b) **"Bluetooth"** module for Windows system and for Gumstix which would send and receive distance data respectively to travel the required amount.
c) **"Motor"** module which would drive the motors using L298N motor driver to make the trash can move for the calculated amount of time.

Implementation Details:

a) Gesture module:

The gesture module is used to recognize the gesture of the person willing to throw the trash. The Kinect Camera is the sensor which we use to accomplish this task. The camera grabs frames at 60 fps and waits for the person to perform the gesture so that the coordinate mapping can begin.

Figure 1: XBox Kinect Camera

Once the person performs the gesture, which is just the sideways raise of hand in our case, we save that frame and use Kinect's in-built libraries to get the coordinates of the hands. Once we get the coordinates of the hand, we use the grabbed frame and do color segmentation on it to recognize the trash can. We made our trash can light green in color to differentiate it from the surroundings. Once the trash can gets segmented, we find the coordinates of the trash can.

This was one of the most complicated parts of the project since the Kinect can recognize a person and give us real world coordinates of the same in 3D space in meters. However, it cannot recognize objects and just treats the frame as any other image you would gather from a normal camera. So, the coordinates we got for our trash can were not the real world coordinates but the coordinates in the color space of the Kinect which were just (x,y) coordinates in pixels. This was a major hurdle for us since we could not find distance between the trash can and the person without the real world coordinates. After digging up online about digital image processing, we found out a way to map camera space points to depth points and using them to find the real world coordinates of the desired object. But then we were faces with another hurdle since Microsoft's official website gave the focal length of the camera in pixels and we needed the focal length in millimeters. We then manually calibrated the camera using various tests and estimated the focal length. Then we used the following formula to calculate the real world coordinates:

$$Y_w/Z_w = Y_v/f \qquad \text{and} \qquad X_w/Z_w = X_v/f$$

Y_w, X_w = Real-World y/x coordinates.
Z_w = real world depth coordinate.
f= focal length

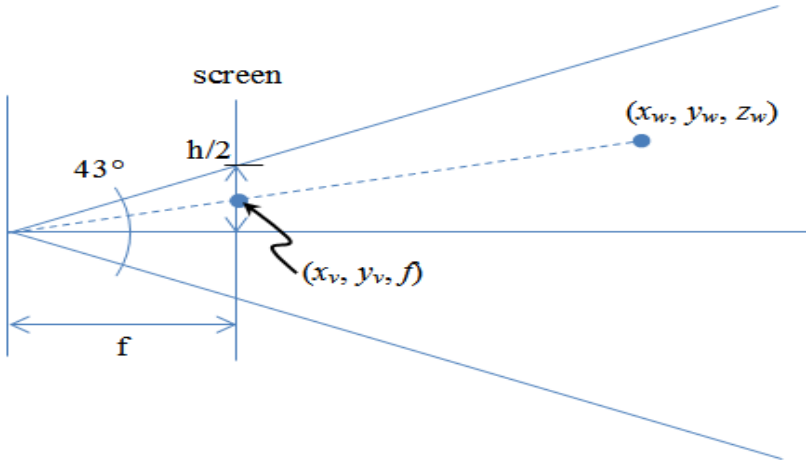


Figure 2: Calculating Real World Coordinates from Pixel Coordinates

We had to introduce a limitation of placing both the trash can and person in the same x-plane so that the depths of both are same.

Once we found out the coordinates, we use the Euclidian distance formula to calculate the distance between the person and trash can.

$$\textbf{Distance} = [(x_2-x_1)^2 - (y_2-y_1)^2]^{0.5}$$

b) Bluetooth module

We implemented the Bluetooth interface using RFComm Protocol to communicate between the Gumstix and the Windows laptop which did the image processing part.

We encountered a lot of problems in this part as well since due to our image processing system being on Windows and Gumstix running on Linux. Microsoft provides a RFComm code but that code didn't work for us. We had to make a lot of changes to that code. Every time we tried to send the data over Bluetooth the Laptop would refuse to find the socket it has to connect to with Gumstix and that created a lot of problems since sample Gumstix code we were provided with had no such thing as socket mentioned in it. Initially, we thought of making the laptop as server and Gumstix as a client since Gumstix was just receiving signals from the laptop. But since that didn't work for us, we did it the other way round and we made the Gumstix poll for the input every 10ms. This solved the problem and we were able to set up a stable connection between the image processing system and Gumstix.

c) Motor module

This was the simplest part of our project since we just needed to drive the motor for a fixed amount of time which was calculated using Time = Distance / Speed formula. We used the L298N motor driver module to provide the motors the right voltage and current from the Gumstix and drove them using 9V batteries. We already knew the speed at which the motor was rotating and we knew the distance as well. So, we just used a timer to make the motor run for a certain amount of time.

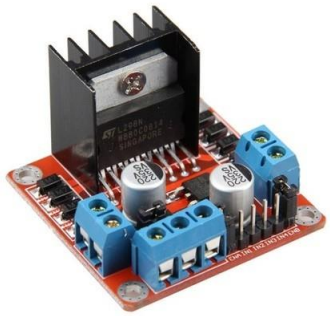


Figure 3: L298N Motor Module

Figure 4: Final Product

IV. Individual Contributions:

Abhishek Gaur: Image Processing using Kinect (assisted by Anubhav)

Anubhav Gupta: Bluetooth Module and Motor Module (assisted by Ritika)

Ritika Chatterjee: Bluetooth Module on MacOS (we had to drop this since the module wasn't working correctly due to compatibility issues of Kinect with the MacOS).

V. Summary

This project involved a lot of key concepts from class as well as made heavy use of computer vision and digital image processing and gives a good example of a field that has been growing rapidly these days, i.e., Embedded Vision. We successfully implemented the individual modules and got them to work in unison over different platforms.

Despite all of this and careful planning, we ran into various problems that were practically impossible to foresee when applying our design to real world problems. However, through robust modeling and frequent design iterations coupled with a few limitations, we were successfully able to design and implement a working prototype of our smart trash can.

VI. References

a) "Trash Can Level: JAPAN". (https://www.youtube.com/watch?v=fEC0FVT8wek)

b) OpenCV (www.opencv.org)

c) Microsoft Kinect Official Documentation. (https://developer.microsoft.com/enus/windows/kinect)