Chez Bot

Project proposal

CSE 291 Human Robot Interaction: Final Project

Winter 2017

Ayush Jasuja Francesco Ferrari Saicharan Duppati
{ajasuja, fferrari, sduppati}@ucsd.edu

I. Overview

The goal of this project consists in creating a robot able to navigate in an unstructured environment, detect humans, approach them and offer them candies through a voice/text interface. If the bot has no more candies then it will enter a “refill” mode where it would interact with a human and ask them to be taken to Chez Bob. We will implement the task by using computer vision algorithms for human detection, PCL and the bumper sensor for obstacle avoidance and the keyboard and the laptop’s speakers for interacting with humans.

2. Algorithmic and Software Components and Task List

The robot will autonomously navigate and use openCV to recognize humans. Once the human has been detected, the robot will approach them and offer a candy bar through an audio message. The human will then have the opportunity of entering “yes” or “no” and the robot will re-act accordingly.

In order to accomplish this task the robot will subscribe to multiple topics:

- The point cloud library measurements
- The raw image from the astra camera for image processing
- The bumper sensor’ state
- The keyboard inputs for input communication

The robot will use the images returned by the camera to detect humans in the area. We will use openni2 combined with RGB-D data to detect the people’s faces and determine their location. While the robot searches humans it will avoid obstacles by using the bumper sensor and the point cloud library.

The robot will interact with humans through a combination of visual and audio outputs such as a pre-recorded message in which the robot offers a candy and a screen asking the human to re-act accordingly, and through text inputs taken from the keyboard where the human will either accept or reject the offer from the robot.

If the robot has no candies it will interact differently, in fact it will approach a human and ask to be taken to Chez Bob to get more candies. The human or the robot will then update the “hasCandies” flag and return to the normal search mode in which offers candies.

The robot will need to remember a total of 16 states defined by four binary flags:

- “detectedHuman” set to True if human is visible
- “detectedObstacle” set to True if obstacle is detected
- “hasCandies” is bot still has candies to offer
- “isInChezBob” is the bot is inside the constrained area of the Chez Bob

We will use a priority queue for this state machine in which obstacle avoidance is our top priority.

2.1. High-level Program Overview Graphic

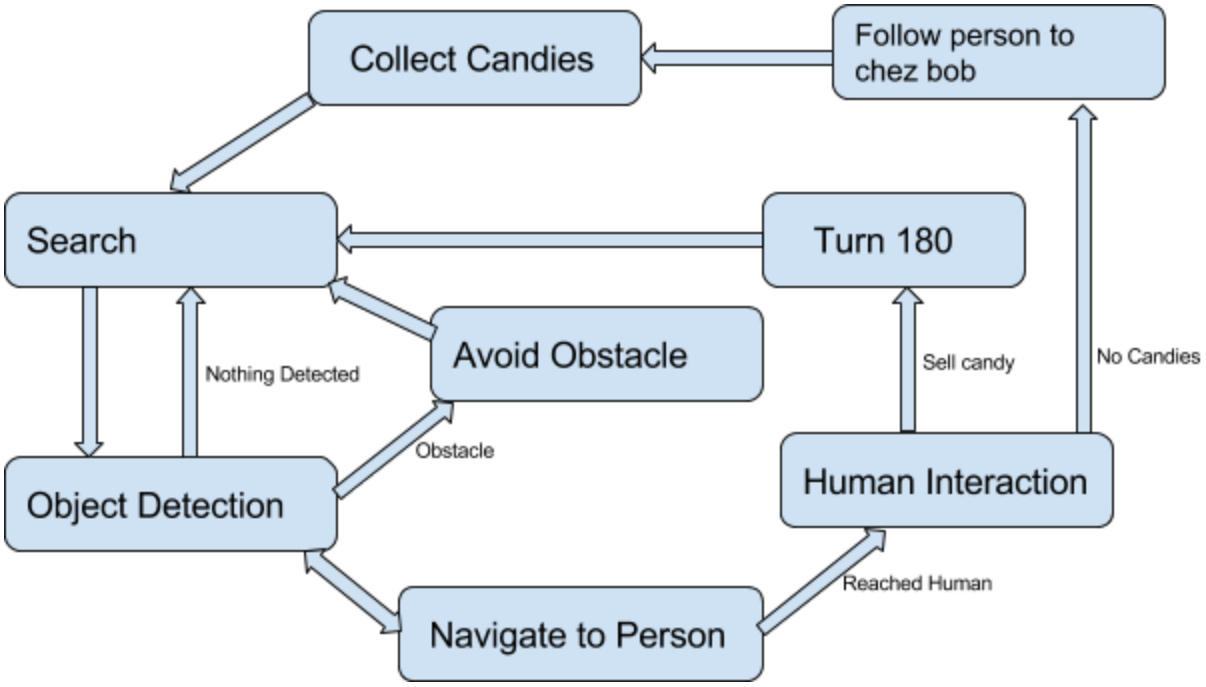

Fig 1: Proposed finite state machine and control flow

2.2 Detailed Textual Description

Chez bot starts from the “Search” state in which it constantly interacts with the object detection module to find objects which come in its way. Once the bot detects an object, it decides whether the object is a human or not. If it finds just an obstacle it navigates to avoid the object and moves back to the search state, where it tries to find more customers to sell candies.
If a human is detected, the robot switches its state to “humanDetected” and starts moving towards them. Again, while approaching a person, it might find some obstacles which it tries to avoid by switching to “obstacleDetected” state and after avoiding them, follows its path towards the person.
When a human is in robots proximity, it tries to sell candy and runs a prerecorded message to offer candies. Robot then open a text interface where user inputs whether they would like to have a chocolate or not. If they enter a positive response, robot waits for ten seconds for the user to collect candy or moves back in case of a negative response. In case no input is received from user, chez bot moves back after waiting for a few seconds.
In case the robot is out of candies, the person detected is asked to take it to chez bob. If they agree to help, the robot just follows them as they move towards the destination. Once in chez bob, robot refills its stock and moves out in search of more customers.

2.3. Task Distribution

Task Name	Description	Task Leader
Detecting Humans	Detect humans based on specific features such as faces and turn on a “detectedHuman” flag	Francesco
Handling Interaction	Stop in front of human, ask for input and receive text input	Sai
Navigation	High level controller for navigation which includes approaching human, search	Ayush
Obstacle Avoidance	Avoid obstacles that are not defined as humans using PCL and bumper sensor	Ayush
FSM	Design overall architecture of the FSM to coordinate between states	Whole team
Refill candies	If “hasCandies” flag is False then follow human to Chez Bob	Undecided

2.3.2 Possible Extensions:

The following are the possible extensions to the proposed project that increase the appeal and the usability of the robot that may be implemented during the project period. Most of them are open-ended design questions which we might want to answer during the same.

1) Keep the count of the candy it sold and once everything is sold, retire to the fixed refilling point (Chez Bob):

Approaches:

1) SLAM
2) Ask the last buyer to take the bot to Chez Bob
3) Recognizing the room door based on the color of the sheet attached to it.

2) Voice input from the buyer instead of keyboard input (Differentiate between 'yes' and 'no' voice input)

3) Handling obstacles in path towards human:

The robot may have to cross multiple obstacles to reach human and during this period, Following things could happen:

(i) The location of the human could be changed

(ii) The human might have left the scene

(iii) The human is still in the same location, but the robot position and orientation may be so skewed due to obstacle avoidance that it doesn't see the human anymore

4) Handling more than one person.

Consider a scenario where the robot detects a person p1 and moves towards him but another person p2 steps in, The following choice needs to be made: Will the robot consider the new person p2 as an obstacle or a buyer? If it thinks it is a buyer, should it give preference to p2 than p1?

5) What is the default path that the robot takes if it sees neither human nor obstacles? Can a human somehow briefly describe the path (via drawing may be)?

6) Using weight sensor to detect that the tray is empty and the refill needs to be done before it approaches any other human

2.4. Experimentation and Testing plan:

Task Name	**Testing method**	**Metrics**
Searching Human	Varying the distances and angles between robot and the human and how it affects the detection rate	Max distance, Duration, Angle, Detection ratio
Human vs Picture of a human	Test with pictures/drawings of human vs actual human	Detection ratios across physical human and pictures between different methods of recognition
Moving towards the human	Varying the distances and angles of approach between the robot and the human	Stopping distance between robot and human
Getting back to the origin (the default position)	Vary the location points from where the robot needs to get back to the origin	Origin wrt to the current state
Interaction between robot and human	Testing on various keyboard inputs; checking timeouts when person stop interacting	Timeout, keyboard keys
Fallback in case of failed interaction	Testing the cases where the robot cannot move towards the human 1) Human not available anymore 2) Human cannot be approached due to obstacles in the way	State of the bot, which needs to be updated in case of a failed transaction
Following the person to Chez Bob during refill	Testing on consent to take the robot to chez bob; Testing the following part	Distance of separation between person and the robot while moving to Chez Bob
Getting out of Chez Bob	Locating the exit of Chez Bob based on the color	Identification of the chosen color under various lighting

	sheet attached to the door	conditions

3. Supply Needs

The project will not involve extra hardware beside the turtlebot and the laptop provided. Speakers might be used if we determined that the output volume of the laptop is not sufficient.

Object	**Have or need?**	**Who has it?**
Speakers	Have	Team members
Laptop	have	Team members

4. Project Gantt Chart and Schedule

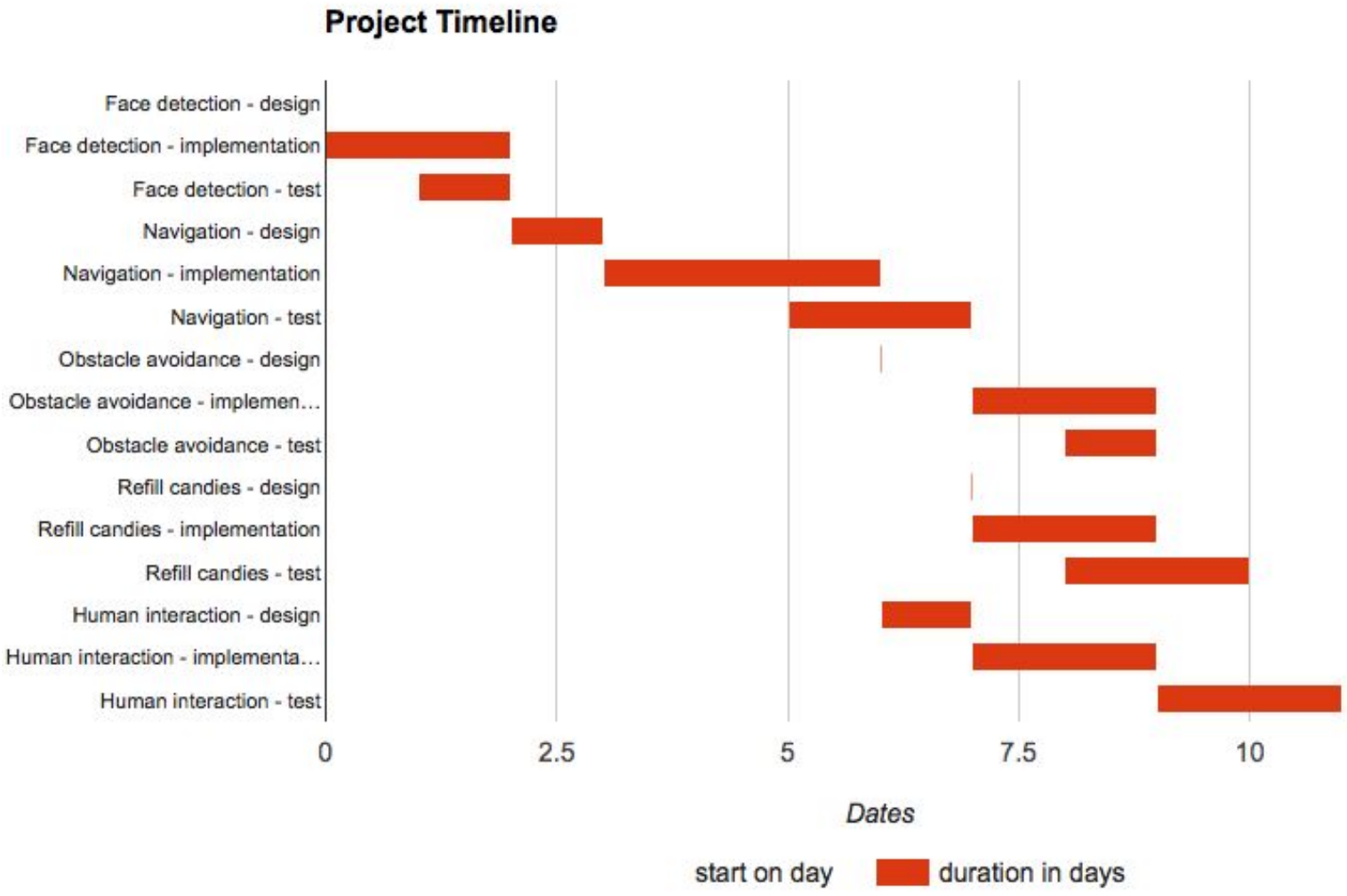

Fig 2: Gantt Chart - the start date (x=0) is March 3rd.