The Crawling Autonomous Robot CAT

Seon Kim UFID:3069-2580

Robot Name: CAT

EEL5666 Intelligent Machine Design Laboratory Final Report

Instructors: Dr. Arroyo, Dr. Schwartz TAs: Mike Pridgen, Thomas Vermeer

Date: 12/08/09

Table of Contents

Abstract

Inverse Kinematics is a popular method for robot motion control and many hexapod walkers have been built using Inverse Kinematics. Despite the fact that quadrapods require a fewer number of legs that need to be controlled than hexapods, quadrapods have been less popular mainly due to the balance issues as well as the difficulty designing dynamic Gait algorithms when dealing with only 4 legs. This paper introduces the Crawling Autonomous Robot (CAT), a 4 legged quadrapod walker that uses Inverse Kinematics and effectively solves these balance and movement issues using Step Interpolation and highly customizable Gait algorithms.

Executive Summary

CATs main task is to search, find, track, attack or interact with red objects. It starts by randomly walking around until the red object is spotted. It then acknowledges that it sees the object by nodding. If the object happens to be too high to reach, CAT will start a stationary tracking behavior by rotating in one spot according to the movement of the object and keep the object in line of sight until the object is on the ground. Once the object is at ground level, CAT will start walking towards the object. If the object is close enough, it enters an attack sequence where CAT raises its front leg and smashes the object. Finally if CAT happens to lose sight of the object being tracked, it will shake its head and return to the random search behavior.

CAT uses the CMUcam3 to track colors. Both the x and y coordinates returned from the camera determine the height, distance, and position of a tracked object. In turn, this data is used to determine CATs behavior. Since a lot of CATs functionalities depend on the CMUcam3, it was important to ensure that the camera was tracking and functioning properly, more so since the CMUcams are well known to have color tracking problems due to lighting.

The other half of CATs behaviors are controlled by the usage of 4 Infrared Range sensors. The sensors determine how close CAT is to a wall and determines the rotation angle, speed, and stride width to execute collision avoidance.

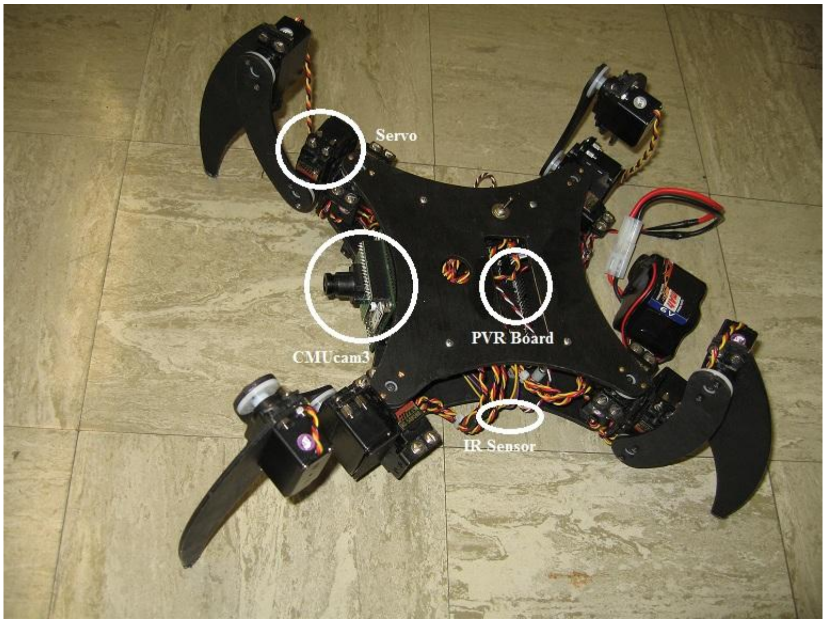
Underlying CATs behaviors are modularized Gait algorithms and leg movement sequences that are executed through the usage of Inverse Kinematics and Step Interpolation. The motion algorithm is designed in such a way that it efficiently takes advantage of the high degrees of freedom that each leg has to offer all the while keeping the movements constrained to a certain limit in order to keep CAT balanced. The only parameters that need to be specified are the rotation angle, speed, stride width or x and y coordinates, and the correct posture or movement is automatically calculated. The high level search, track, and attack behaviors presented in the current version of CAT are only a small example to demonstrate the capabilities of this algorithm that CAT runs on.

Introduction

Instead of building a robot with impractical or useless tasks that will have no use in the real world, or menial tasks that operate at suboptimal levels, CAT will take advantage of and improve upon what robots are naturally do: fascinate and entertain humans. CAT is an interactive behavioral robot that interacts with its perceived environment in an entertaining way and expresses itself with various motions. The CAT currently has motion control implemented largely through the usage of inverse kinematics and collision avoidance using 4 IR sensors.

Integrated System

CAT is a 4 legged walking robot with its focus on behavior, motion, aesthetics and ease of use. The individual components of CAT are shown below.

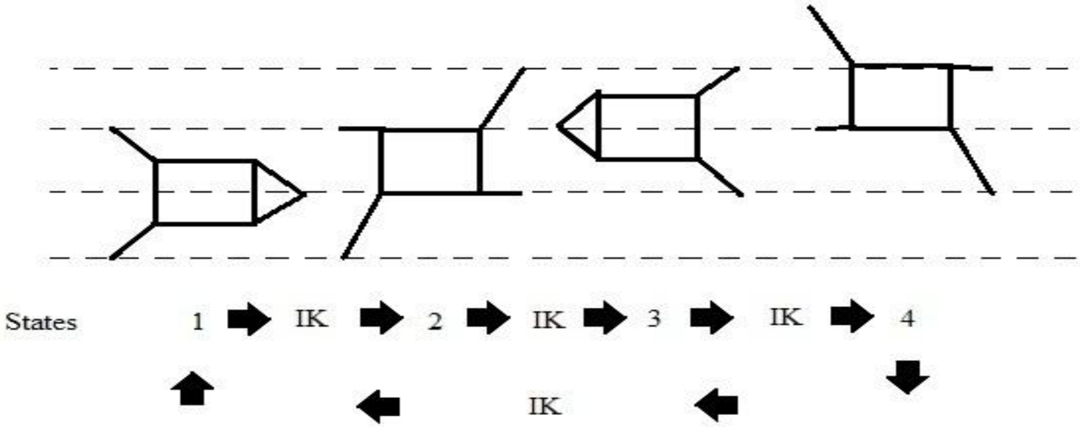


The PVR Board is mounted in the center of the body and the CMUcam3 is mounted in the front. Although the picture shows a larch 3300mAh battery, a smaller 2200mAh battery pack can be mounted inside the robot when actually running.

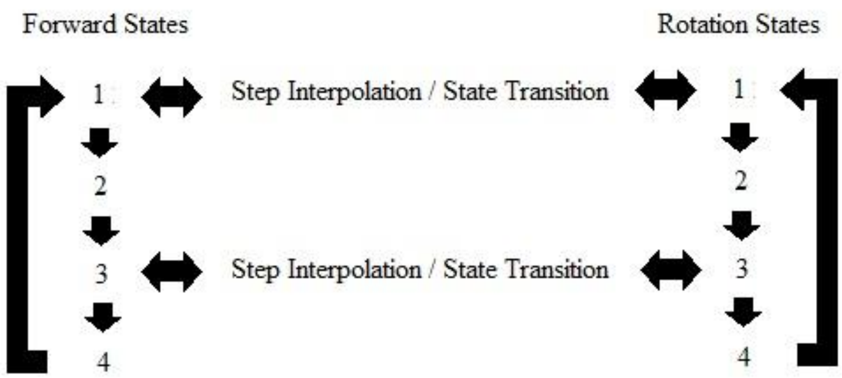
Actuators

The prototype robot uses twelve HS-5485HB Digital servos with three on each of its four legs. The high degrees of freedom these servos provide enable great maneuverability and allow a more accurate means to exhibit and represent its behavior through postures and actions.

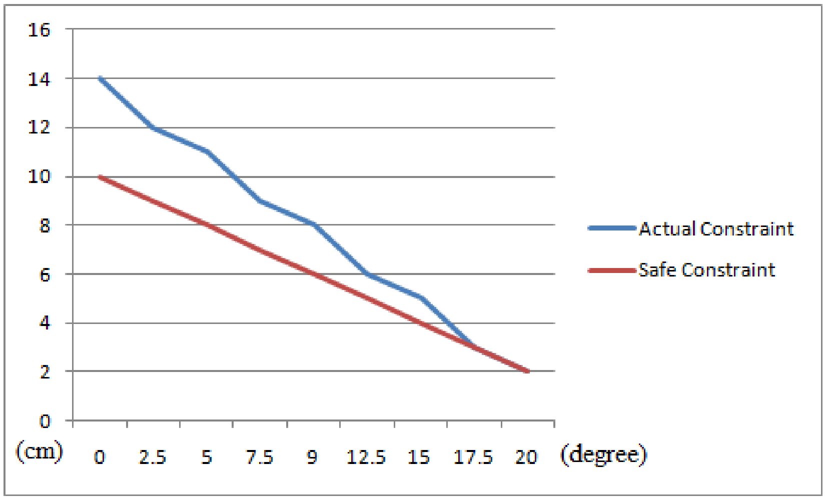
As of now the motion is divided into mainly 4 states and transitions between these states are done with inverse kinematics. These states are shown below.



States 1 and 3 are the same for all motions and these states are used for transition. Below is an example of how this works between the forward and CCW rotation states.



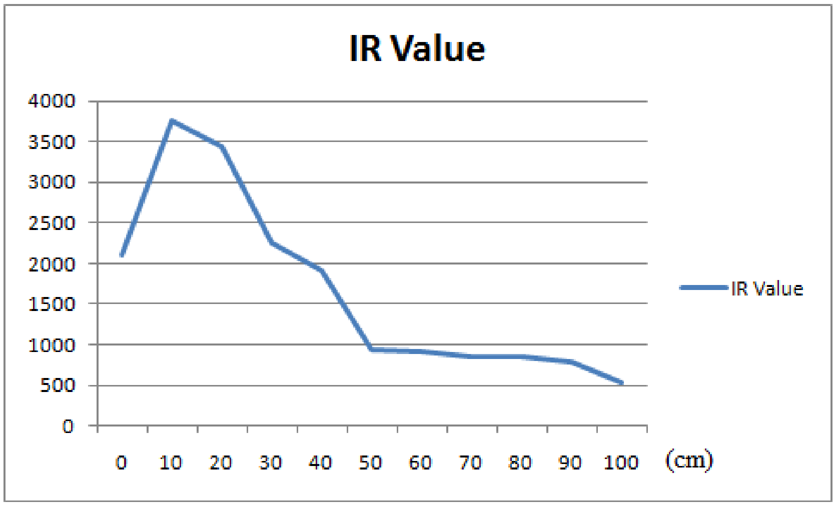
Each position of the legs between the states are interpolated and calculated by inverse kinematics. This allows a large degree of freedom for the legs, and at any given state at least one leg will be in its correct state position. This also allows the parameterization of the rotation angle and step width, and these parameters can be any floating point value within the constraints of what the robots body can allow. The constraints were plotted out through experimentation and a safety limit was put in place to ensure that the robot operates smoothly. This is shown below.



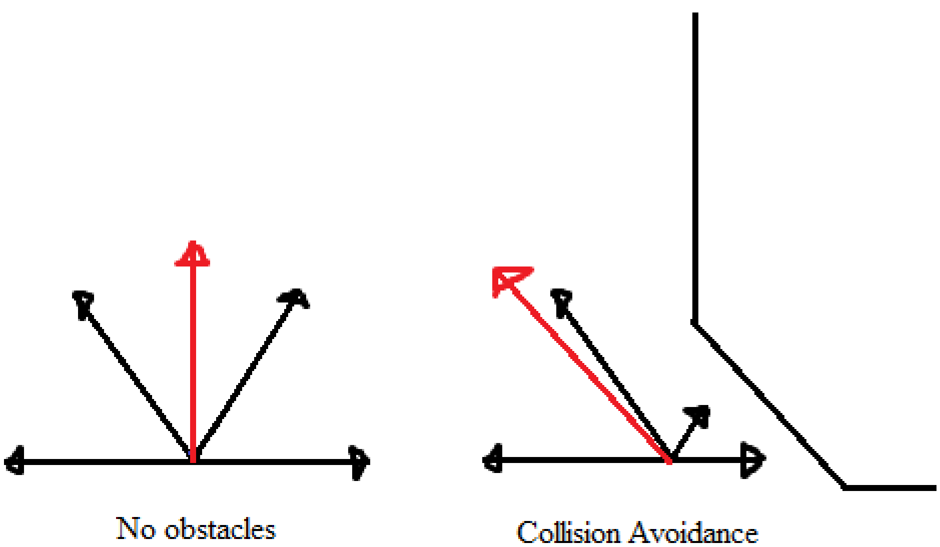
Another important feature is that the IK and interpolation is done through interrupts for correct timing. This allows speed control of the movements and the robot can move and slow speed without being too jerky because of sparse interpolation. Another advantage of this is that it leaves the main loop empty and open for other operations CMUcam data processing.

Sensors

4 IR sensors are mounted on the bottom of the robot to do collision avoidance. At each motion state, the IR sensors are read in multiple times and the average is taken and linearized by the equation Range = 1/Voltage – 4, then added by treating them as a vector. Below is the sensor readings plotted out without linearization.

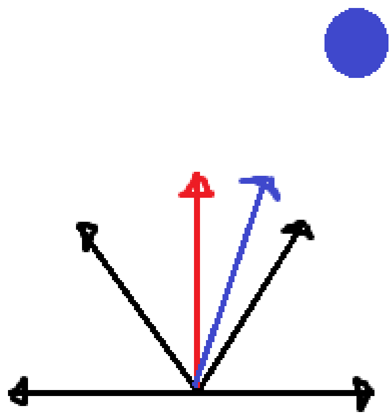


The rotation angle and step width of how the robot moves to the next state is determined by adding the IR vectors together and ensuring that it stays within the constraints that were explained in the previous chapter. Below is an example of this collision avoidance.



Above, the black vectors are the IR sensor values and the red vector represents the direction and width the robot will move to next.

Using just IR sensors, the robot only moves forward when no obstacles are detected. A CMUcam3 is used to do color and motion tracking and the movement of the camera will be controlled by shifting the body of the robot in the correct direction. This will add an additional 5th vector to give the robot a sense of direction when not avoiding obstacles. This is demonstrated below.



Object Tracking

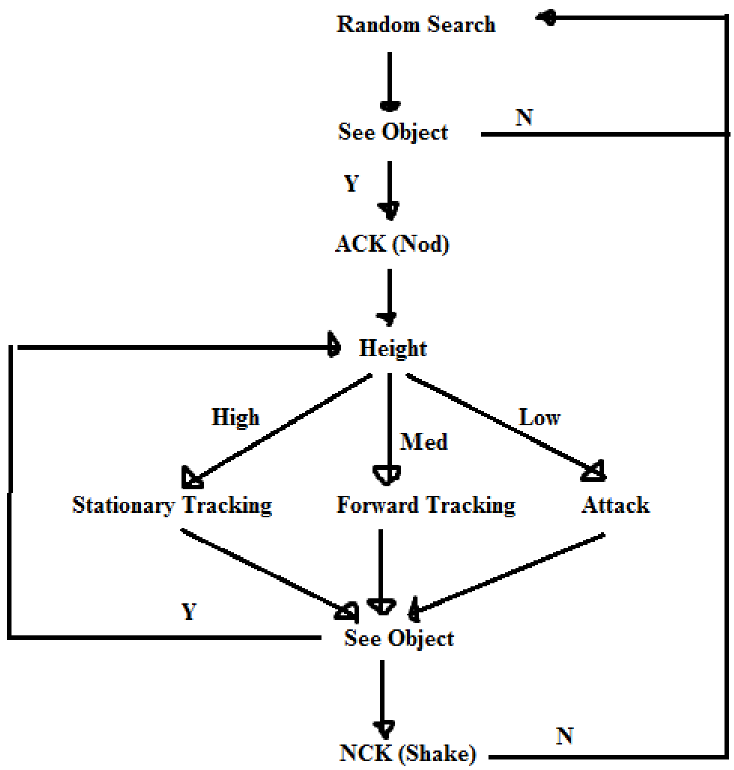
The y coordinates of the CMUcam3 also determines the height of the object. If the object is too high, CAT will do stationary tracking. If it is at ground level, it will move towards the object. Finally, if the object is too low, this signifies that the object is close within CATs reaching distance and CAT will attempt to attack it.

The table below shows a summary of the sensors used.

Sensor	Number	Task
CMUcam3	1	Color and motion tracking
IR sensors	4	Object and edge detection

Behavior

The behavior is coded on the Pridgen-Vermeer Robotics Board. On top of maneuvering through its environment and avoiding obstacles with four legs, the robot acts upon the object it is tracking. The flow chart below depicts the high level behaviors implemented in this version.



Conclusion

Despite not knowing anything about robots or how to build them when joining IMDL, CAT turned out to be quite a success. The only limitation was the amount of time I had since I had to learn everything from scratch. Another minor problem was working with the CMUcam3 since most of CATs behaviors depend on it and it is known to have lighting issues when it comes to color tracking. Other than this, the project went smoothly. I would not change anything even if I were to start over. For future work, now that I do have the time, I plan on adding more sophisticated behaviors to CAT.

Documentation

Pridgen-Vermeer Robotics Board

CMUcam3 (http://www.cmucam.org/)

Sharp IR sensor GP2D12 (http://www.acroname.com/robotics/info/articles/sharp/sharp.html)

HS-5485HB Digital Servo (http://www.hitecrcd.com/servos/show?name=HS-5485HB)

Appendices

The code consists of 5 enormous files and could not be listed here. Please visit for the http://sites.google.com/site/projectcrawlbot/ complete code listings