Satrap @Home Team Description Paper

Behnam Azimi, Behzad Azimi, Nima Hazar , Alireza Bagheri ,Mohammad Kashani, Shahin Mohammadi, Alireza Mohammadkhan, Hossein Mohammadkhan, Ali Tayyebi , Amin Yazdani and Pedram Zohouri

Satrap Group Tehran, Iran
and Ca,USA

{Azimi, Behzad, Alireza, Kashani, Amohammadkhan, Mohammadkhan, Tayyebi, Zohoori }@Satrapteam.ir http://www.Satrapteam.ir

Abstract. This paper is aimed to summarize the overall process which is done by Satrap team in order to prepare for participating in Robocup @Home competitions 2008, including hardware and software specifications of our two robots. For the second year competing, this year researches are focused on optimizing last year both hardware and software systems and also designing new methods to cover additional abilities for new testes. Our goal is to build an interactive humanoid robot which could act autonomously and be used in indoor environment to help people in their daily life.

1 Introduction

Robocup is an international joint project to promote AI, robotics, and related field. It is an attempt to foster AI and intelligent robotics research by providing a standard platform where wide range of technologies can be integrated and examined. Specifically the aim of Robocup @Home competition is to use the development of useful robotic applications that can assist humans in everyday life. To reach above goal the competitions consists of a series of specific tests and is being held in a home-like indoor environment. For this year we have designed and implemented two autonomous robots, named C3PB and R1D2 which are reminiscent of Star War's C3PO and R2D2 and we are trying to get to their abilities and features too! R1D3 is the improved version of our last year's robot R1D2 and R1D2 and is especially designed for tracking and navigation. C3PC is built up based on C3PB and C3PA platform, equipped with two hands with 3 degrees of freedom consisting of 5 fingers, which are appropriate for general home needs.

Many researches have been done in the recent years in order to develop home robots [1] [2]

2 Hardware

In designing our robots we placed great emphasis on modularity and extensibility so, although two robots have specifically different mechanical and electrical

Structure, but some common base modules are used in both. Also the power supplying methods are the same. In this section we are intending to briefly explain hardware specification of C3PC and R1D3.

2.1 R1D3

R1D3 has about 80 (cm) heights and about 35 (cm) widths. R1D3 main processor is a BUGbase. it is consists of It's a fully programmable and "hackable" Linux computer, equipped with a fast CPU, 128MB RAM, rechargeable battery, USB, Ethernet, and a small LCD with button controls. It also has a tripod mount because, well, why not? Each BUGbase houses four connectors for users to combine any assortment of BUGmodules to create their ultimate gadget. Detailed specifications of these modules are in Table 1

Main sensors that are connected to Notebook are an omnidirectional USB camera, a direct Firewire camera and 4 USB web cameras. The omnidirectional and those 4 USB web cameras are connected to Notebook via USB Selector. And the direct camera is connected to Notebook with IEEE1394 Protocol

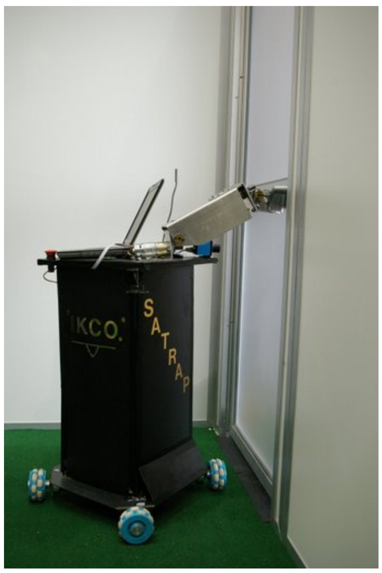
2.2 C3PC

C3PB has about 170 (cm) heights and about 65 (cm) widths. Its main processor, like R1D2, is a Processor Board which manage behavior of robot, consists of two BUGbase And 2 motion detector. Detailed specifications of these modules are in Table 1

- ARM1136JF-S-based microprocessor
- 1 USB 2.0 HS host interface/4 hub port connections
- 1 USB OTG HS interface
- 4 UART serial links
- 4 channel SPI interface
- I2C (400 kbits) interface/4 channels
- I2S interface/2 channels
- Smart LCD interface
- Camera sensor interface
- Micro memory card interface
- MPEG4 hardware encoding/decoding
- Hardware graphic acceleration
- 10/100 Ethernet MAC
- Base unit LCD module interface
- Base unit onboard memory (FLASH/DDR SDRAM)
- JTAG/ICE support
- Serial debug port
- Power system
- AC operation
- Battery operation/up to 4 external batteries
- Fast battery charging/simultaneous of internal and external batteries
- Smart power management support
- Battery-backed real-time clock
- Audio out via onboard piezo speaker
- GPS

Table 1. Results

For C3PC, as sensors, there are 2 Firewire cameras which extract depth of the scene, just like what human eyes do. These 2 Firewire cameras connect to the Notebook via IEEE1394 Protocol.



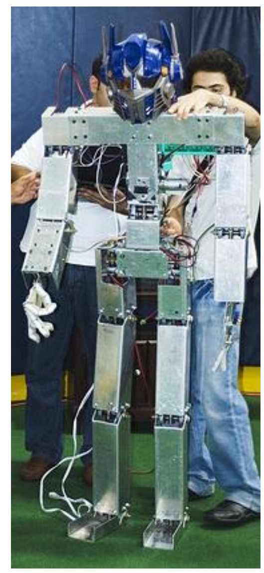


Fig. 1. R1D2 Fig. 2. Behnam, C3PB, Amin


Fig. 3. C3PB Fig. 4. R1D2 boards

3 Software

Our last year robot software design was Windows-based and was based on Microsoft Visual Studio .Net, but This year we transform our works to Linux environment and develop our programs by C++, using GNU C Compiler. In this section we have briefly described our solutions to different challenging problems which are involved in @Home competition and also explained our implementation methods.

4 Localization

Localization is a major area of mobile robotics, which lots of efforts are focused on it. There are several classic approaches to this problem [?]. In general there are two basic types of localization:

Global Localization Determining position of the robot under global uncertainty. This problem arises, for example, when a robot uses a predefined map and is not informed about its initial position in the map.

Position Tracking Estimating location of the robot while it is moving. The problem arises from errors in robot's odometry which can ruin the estimation of robot location.

In designing our robots we used a novel statistical approach, which is based on color distribution of the landmarks.

4.1 Feature Extraction

The robot needs to identify distinctive places which it has to visit, in order to distinguish vertices in our topological map. Distinctive places are defined in terms of sensory measures, which are also known as distinctiveness measures. It promises that the robot will converge to these distinctive points from anywhere within a local neighborhood, simply using a hill climbing method without the problem of falling into local minima.

We used an omnidirectional vision sensor on our robot to capture 360 degrees around, and defined the statistical distribution of top colors around each place as the d-measure function. Top colors are colors which have the most frequencies in the captured image. In other words, our feature dictionary consists of top colors and their cumulative distributions. In order to compare current state with the stored states, we extract the cumulative distributions of desired state top colors, and compare them with their expected value from dictionary. In order to have the ability of adjusting the accuracy and measuring error percentages of this method we systemize our measurement by using chi-square distance.

Significance of the chi-square test for goodness of fit value is established by calculating the degree of freedom v, and by using the chi-square distribution table. The v in a chi-square goodness of fit test is equal to the number of categories, c, minus one (v = c -1).

4.2 Global Localization

In this part we present our approach to find the initial position of our robot. As mentioned before, we have been created a dictionary of distinctive places beforehand, which is used by our algorithm. Our goal is to find the nearest distinct place located on the Voronoi graph and send robot towards it. To achieve this goal, first we transform images from our omnivision sensors into linear space [3, 4]. Then we calculate the chi-square distance of the current position to each place stored in dictionary, and find out the minimum distance. Finally we offer a distance and a direction to the control layer of the robot, which is a vector calculated from averaging the distances between center of mass of current and destination top colors.

4.3 Position Tracking

Position tracking is fairly simple using chi-square test. Since we know the destination vertex and its color distribution, we could simply calculate the chi-square distance between current and destination position, and try to minimize it using hill climbing method. In other words, on each movement we want to minimize the measure function by following the Voronoi edge towards next vertex and use compass sensors to mitigate the motion errors. This process continues and after each step we test the goodness of fit, in order to recognize that we reached the neighborhood of destination vertex.

5 Path Planning

There are many algorithms for configuring space; one of the common algorithms which we used for building roadmaps is Voronoi Graph approach [5, 6]. Because In the plane, the Generalized Voronoi Graph (GVG) simply equals to GVD and both are the set of point's equidistance to two obstacles in the environment; so we use GVD for our path planning. After representing GVD by finding its edges and vertices we curve our path from source to goal position on the Voronoi arcs in order to have maximum distance from obstacles. Finding path from the initial position to the nearest vertex on GVD and also path from last GVD's vertex to goal position is finding by a hill climbing algorithm using our localization algorithm.

6 Obstacle Detection

For being assure that the robot, through its path, has maximum distances from obstacles, we try to move on our GVD's edges. So by using data which is got from a compass sensor, we try to correct any possible error on robot's path that is happened because of its locomotion noises. This mechanism is profit for avoiding collisions with static obstacles, but for avoiding bumping to dynamic or even moving objects we wrote another program, which its thread is executed parallel by our robots locomotion on the planned path. This collision avoidance

Procedure uses data from another direct camera which capture images direct to robots motion direction and if it detects any objects on the path, it will update the map which is built from the environment and call the GVD builder procedure again and update the path.

7 Human Robot Interactions

HRI is the study of interactions between people and robots. It is multidisciplinary with contributions from the fields of AI, robotics, NLP, machine vision, cognitive science and so on. In design phase of our robot, we have been considered two kinds of interactions, via voice and gesture.

7.1 Speech Recognition

It is the first mechanism we considered as the primary communication gate for our robot. The robot has a speaker independent speech recognition system which is based on DARPA-funded Sphinx engine, and is able to take commands from its owner via microphone.

7.2 Gesture Recognition

A primary goal of gesture recognition research is to create a system which can identify specific human gestures and use them to convey information or for device control. Approximately in the year 1992 the first attempts were made to recognize hand gestures from color video signals in real time. We just started working on developing mechanisms for hand gesture detection, by extracting skin-like pixels of the image and try to figure our it's structure by comparing it with some stored gesture from its dictionary.

References

1 Wei Gao, Amy lee, R.m.M.T.: Avateur ii : Home robot system. INDE 599 Final Report, Washington University (2001)

2 : An open architecture real-time robot controller (2002)

3 da Cruz Pinto Gaspar, J.A.: Omnidirectional vision for mobile robot navigation. Master's thesis, UNIVERSIDADE TECNICA DE LISBOA (2002)

4 Wei, S.C., Yagi, Y., Yachida, M.: Building a local floor map by use of ultrasonic and omnidirectional vision sensors. Journal of Advanced Robotics 12(4) (1998) 433–453

5 Choset, H., Burdick, J.: Sensor based planning, part i: The generalized voronoi graph. In: Proceedings of the 1995 IEEE International Conference on Robotics and Automation (ICRA '95). Volume 2. (1995) 1649 – 1655

6 Foskey, M., Garber, M., Lin, M., Manocha, D.: A voronoi-based hybrid motion planner for rigid bodies (2001)