"BABY BOT," A CHILD MONITORING ROBOTIC SYSTEM

Yanfei Liu, Christole Griffith, and Parul Reddy

Indiana – Purdue University Fort Wayne, Indiana; Email: **liu@engr.ipfw.edu**

1. INTRODUCTION

For over eight decades, robotics has been quite a high tech topic in people's mind. Today though robots are widely applied in industry, military field and even in space explorations, they still rarely impact on people's life. Thus, in order to bring robots close to people's life, a senior design team in the department of Engineering at Indiana – Purdue University Fort Wayne (IPFW) is building a robotic prototype system, called "Baby Bot," to assist parents in child monitoring. This is a two-semester project carried out by two senior electrical engineering students. Our team has finished the design of the system in fall 2005, and they just started to build the platform and conduct the experimental testing in spring 2006.

In this "Baby Bot" system, the robot should be capable of finding and following a 7-10 month old baby in a limited space. The system also shows some artificial intelligence features to determine the danger level that the child might run into and take actions accordingly. For example, if the baby is too close to the boundary of a safe area, the robot will try to distract the baby first. Then, if the distraction fails, the robot will activate the alarm system and notify the parents. The whole system design includes the distraction board and alarm system, robot setup, image processing techniques and robot path planning algorithms. The experimental prototype of this system will consist of a Khepera robot, a scaled down table, which simulates the room, with a total area of 32 square feet and a mocked baby. Our team will use LabView for the software development.

The remainder of this paper is organized as follows. In Section 2, we describe the hardware design of this prototype system. In Section 3, we introduce the software design of this system. Finally, we conclude the paper in Section 4.

2. HARDWARE DESIGN

2.1 The prototype

Figure 1 shows a diagram of our prototype. Our prototype system consists of a Khepera robot, a host computer, the distraction/alarm circuitry and a testing table. The host computer is used to process the image data and then transfer the result to the robot. The Khepera robot will communicate with the host computer through a serial connection. The host computer is connected with the robot through the interface/charger module using a standard RS232 cable. The interface/charger module converts the RS232 signal into S serial signal to communicate with the robot. On the other hand, the robot will activate the distraction/alarm circuitry using a wireless transmission sent to the radio base. In the rest of Section 2, we will detail the testing area, distraction/alarm circuitry design and the robot setup separately.

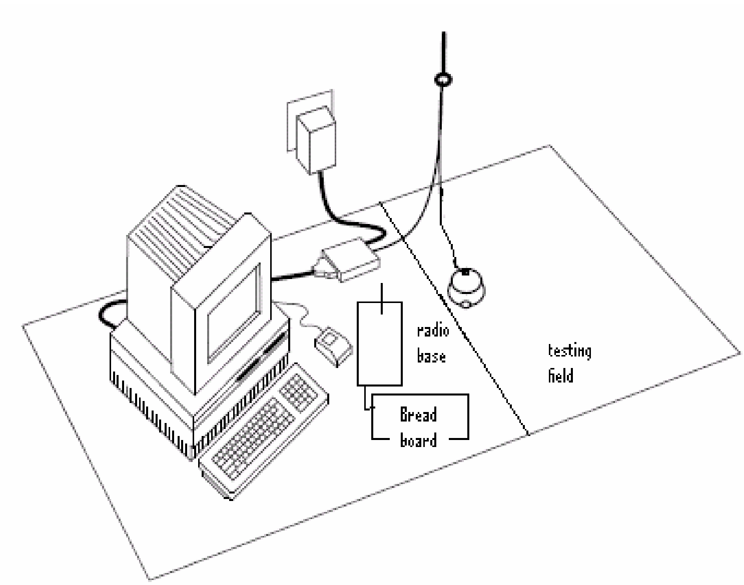

Figure 1: The robot and host computer configuration

2.1 Testing area

A testing field will be built to simulate a baby's playpen area. The field will be a table with approximate dimensions of 1.2m X 2.4m (4ft. X 8ft.). This will act as a scaled down model of a real area. The table surface will be smooth and white. Also, there will be a red marking to indicate the out of bounds area and a blue marking to indicate the warning area. The red marking encompass the playpen and has a total surface area of 1.3m^2 (14sq.ft.). The blue marking will be 5cm to the inside of the red line and will also encompass the playpen area. Figure 2 shows a picture of the testing area. Additionally, to provide a wall, the table will have a white vertical boundary, of height 0.3m (1ft) on its perimeter.

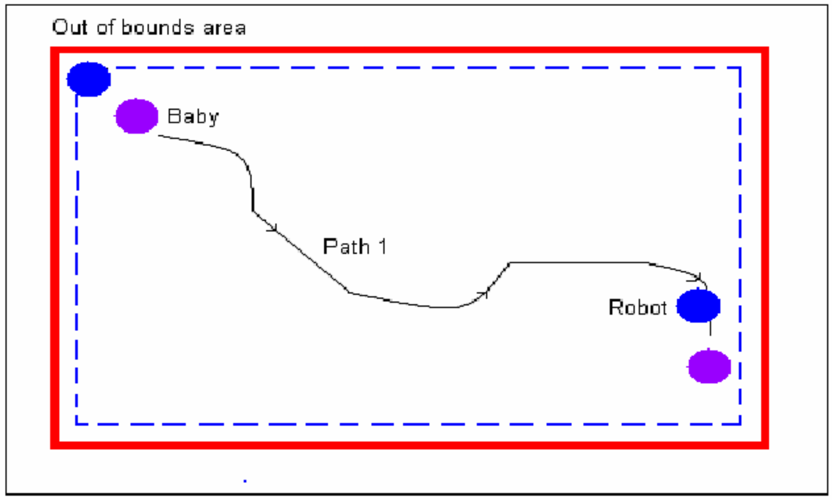

Figure 2: Prototype Design Environment

2.3 Alarm systems and artificial intelligence design

If the baby is approaching the out-of bounds line (blue line) or about to enter into a dangerous situation, the robot will trigger the distraction system (Figure 3 illustrates the design for the distraction circuitry). By doing this, the robot need to transmit an RF signal to the radio base. Once a signal is received, the processor on the radio base will determine to send a signal through J1 to the breadboard. Then the LEDs will lighten up in a certain pattern to distract the baby.

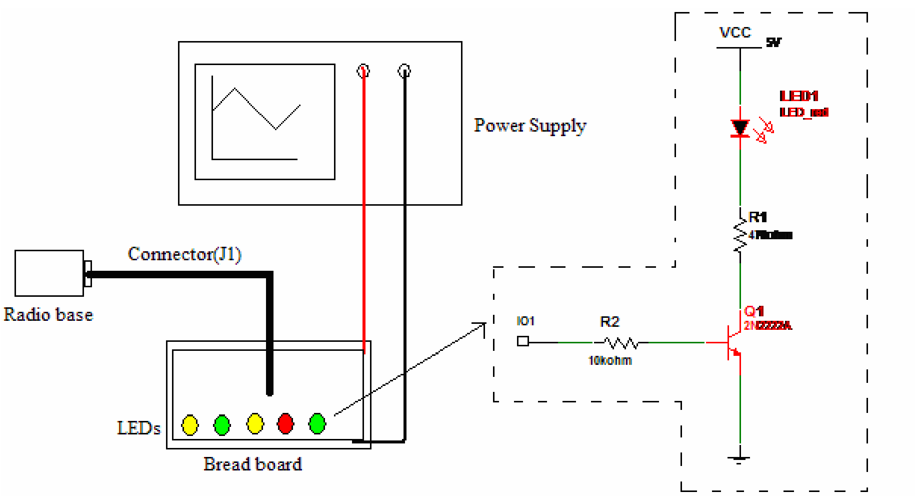

Figure 3: Distraction Circuitry

If the distraction strategy doesn't work and the baby passes the blue margin, the robot needs to sound the alarm system. This way the parents have enough time to respond, before the child is seriously harmed. The circuit (Birdie Doorbell Ringer, 2005) shown in Figure 4 will be implemented in this alarm system. Again, the I/O pin 1 from the processor present in the radio base will activate the alarm whenever needed. The alarm system used is a door bell kit that comes with a solid chime, one lighted push button and a 16V transformer.

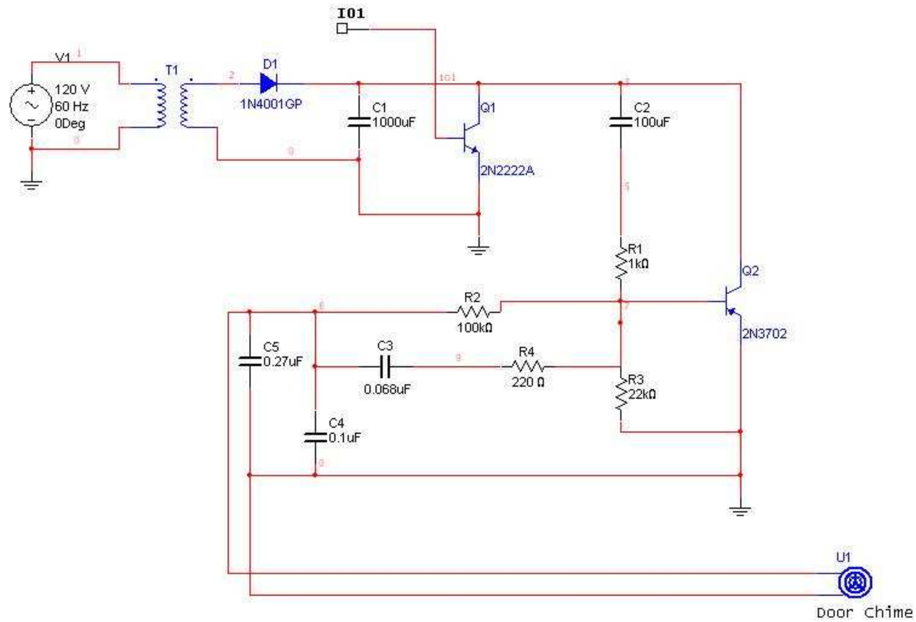

Figure 4: Alarm System

2.4 Robot setup

In our prototype, the robot is the Khepera II (K-team home page, 2005). This robot is compact and has powerful microcontroller capabilities. The robot will be assembled with the following turrets, mounted in the given order, from bottom to top: Base unit, General I/O turret, Radio turret and the Camera turret. To attach these turrets to the robot base, they have to be placed on the extension connector with all the pins seated correctly.

2.4.1 Base unit

An overview of the base unit is shown below in Figure 5.

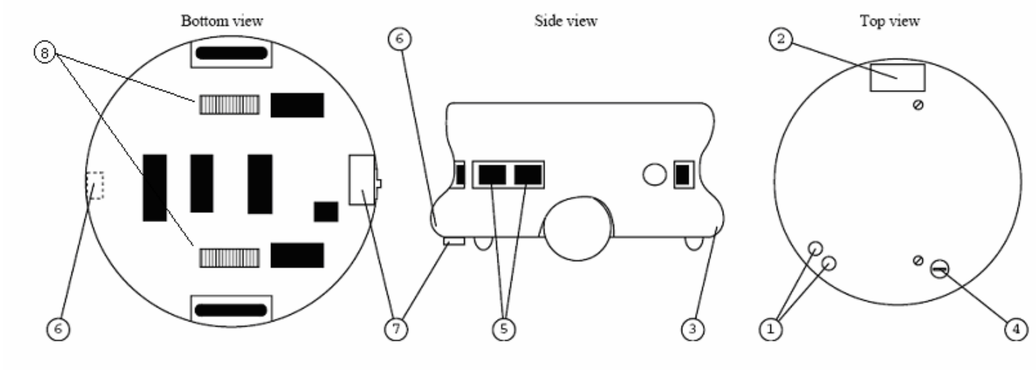

Figure 5: Base unit

2.4.2 General I/O Turret

The next component to be added will be the General I/O turret shown below in Figure 6. This turret will be used to incorporate the distance sensor (used to keep a safe distance between the robot and the baby) into the system.

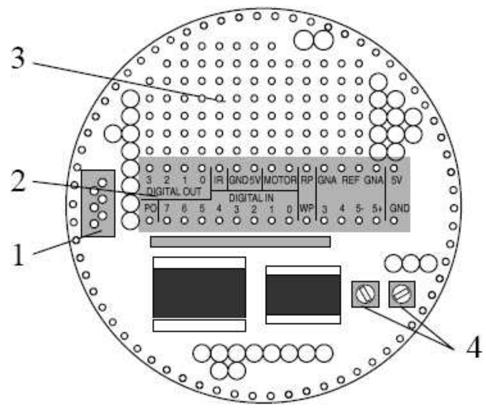

Figure 6: Overview of the turret layout

2.4.3 Radio Turret

An overview of the radio turret is shown below in Figure 7. The serial connection of this turret will be used to communicate with the host computer.

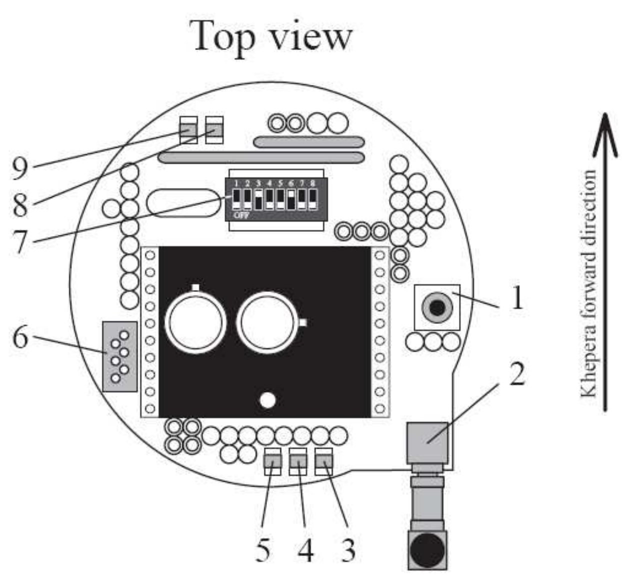

Figure 7: Overview of the turret layout

2.4.4 Camera Turret

Figure 8 shown below gives an overview of the K6300 Camera Turret.

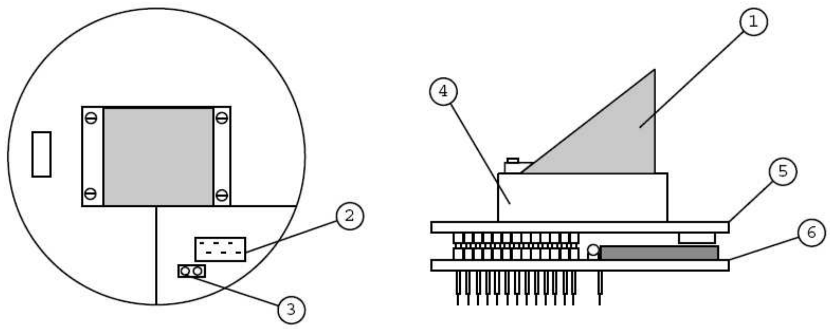

Figure 8: Camera turret

The camera turret comes equipped with a MC68331 processor, along with flash memory and a reset button. The camera turret holds a V6300 digital CMOS color camera and its optical elements. The color images acquired through this camera are 160 pixels wide and 120 pixels high.

3. SOFTWARE DESIGN

In this section, we will first give a brief overview of the overall process (Figure 9). Then, a more detailed description about the algorithm of finding the baby, vision based depth perception and the cellular automata path planning will be given.

3.1 Finding the baby

The camera turret of the robot will be used to find the baby. Once an image is taken, the processor will scan through the matrix until a yellow pixel (under the assumption that the baby wares yellow clothes) is found. Once a yellow pixel is found a virtual flag will be activated. The robot's software application module (Khepera II programming manual, 2000) contains a set of 64 system flags that can be read and changed from either a LabView program or a turret user application.

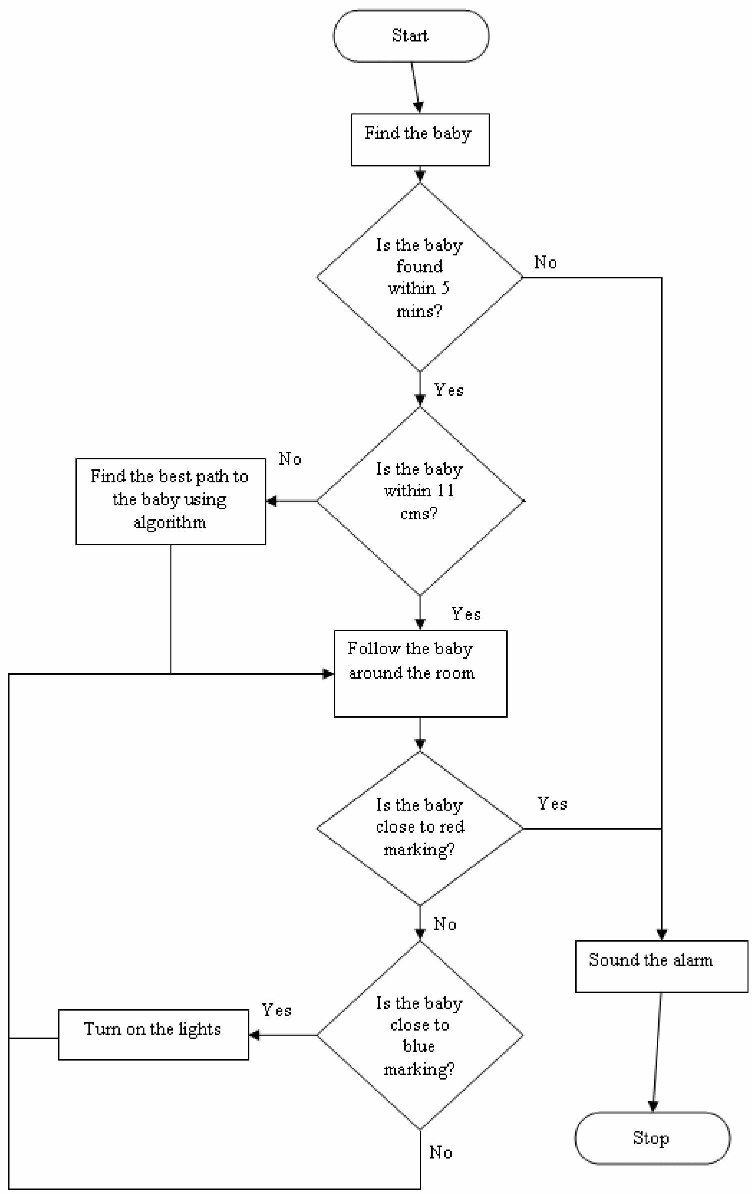

Figure 9: Flowchart of Project

3.2 Vision based depth perception

Sometimes the distance sensor couldn't provide the distance between the robot and the child if the obstacle gets in the way. Thus we need to generate another method to calculate the distance. It is well known that one planar image is not adequate to identify the position of objects in the world coordinates. Therefore we need to use the vision based depth perception method to solve this problem. The depth perception algorithm from an image will be accomplished under LabView.

This processing includes the use of the threshold based method which employs high pass filter magnification of the object boundaries in the image data (Köse and Akin, 2001). This way the edges of an object could be easily detected. A mask of [-1 -1 1 1] is applied to detect rising edges and another mask of [1 1 -1 -1] is applied to detect the falling edges. The output of the application of the masks to all of the image data points is zero or a small number when the image is flat and becomes a relatively high number in case edges are present in the image. These edges are the boundaries of the objects in the image and are represented as peaks in the filtered image. So a rising edge, which is represented by a positive peak, shows the starting point of an object, and the falling edge which is represented by a negative peak symbolizes the end point of the object. As the color difference between the objects become higher, the height and sharpness of the peaks between them also becomes higher. After the detection of the edges, a threshold value is applied over the image to eliminate the false edges, caused by the noise.

This will allow the robot to know the starting and ending points of the objects in the image. The host computer will have a preexisting image of the testing field with which this new image will be compared (Figure 10). As a result, the baby's position relative to the obstacles on the field is determined. Thus, the robot will have an idea of where the baby is located as compared to the other obstacles on the table.

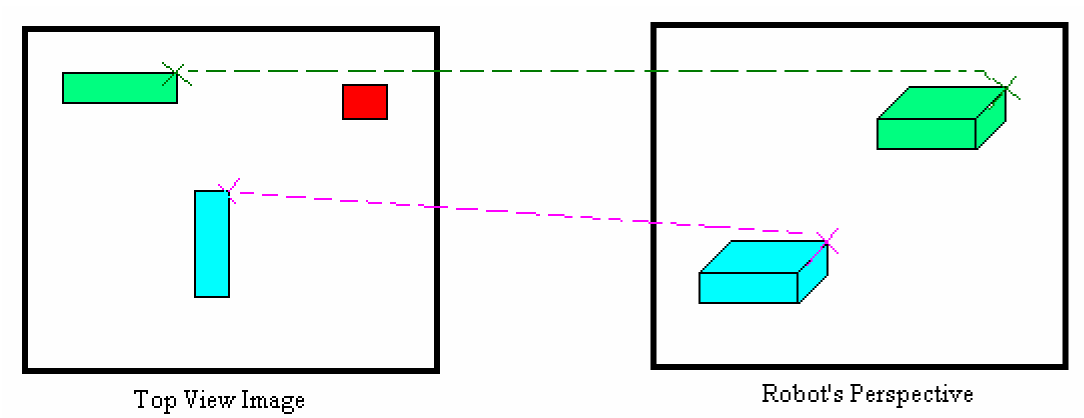

Figure 10: Correlation between different image perspectives

3.3 Path planning: Cellular Automata algorithm

After the baby is found, the robot needs to find a path to move to the baby. In order to solve the path planning problem, the pre-captured image has to be divided into small cells (Figure 11). Each cell is given a number based on the following: 1-Free cell, 2-Obstacle, 3-Goal, and 4-Start.

Thus, once the space has been divided onto grids, the robot could then find the shortest distance to the goal while maneuvering itself around the obstacles (Dijkstra's algorithm, 2005). To do this an adjacency graph will be built (Latombe, 1991). The nodes of this graph will be free cells and the edges will be connections between two free cells.

Therefore, the robot will never intersect with a cell that contains an obstacle. It will always move to the midpoint of a free cell. From this adjacency graph the best path will be chosen to reach the goal. Given below in Figure 11 is a sample path taken by the robot to get close to the child.

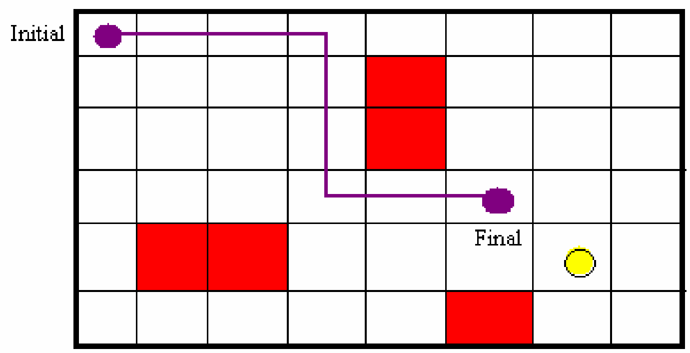

Figure 11: Sample path planning output

4. CONCLUSION

In this paper, we proposed a robotic child monitoring prototype system, which is a Capstone senior design project in Department of Engineering at Indiana – Purdue University Fort Wayne. The whole system consists of a Khepera robot, a host computer, the distraction/alarm circuitry and a testing table. We described the design of each component in details in this paper. Human interactive systems are always challenging, therefore our system is only a prototype which aims at shredding some light in the child monitoring robotic system.

REFERENCES

Birdie Doorbell Ringer (2005), http://www.uoguelph.ca/~antoon/circ/birdie.htm, last accessed on 11/30/05

Paul E. Black, "Dijkstra's algorithm", http://www.nist.gov/dads/HTML/dijkstraalgo.html

K-team home page (2005), http://www.k-team.com/

Khepera II programming manual (version 1.1, March 2000), K-team http://ftp.k-team.com/khepera/documentation/Kh2ProgrammingManual.pdf

Köse H. and Akin H. L. (2001), "Object Recognition in Robot Football Using a one Dimensional Image", *the Tenth Turkish Symposium on Artificial Intelligence and Neural Networks* (TAINN2001), pp. 291-300

Latombe, J. (1991), *Robot Motion Planning*, Kluwer, Norwell, MA, USA