Social Behavioral Robot that Stands in Line

Y. Nakauchi and R. Simmons
Computer Science Department,
Carnegie Mellon University,
Pittsburgh, PA 15213, USA
E-mail: {nakauchi, reids}@cs.cmu.edu

ABSTRACT

Recent research results on mobile robot navigation systems make it promising to utilize them in service fields. But in order to utilize the robot in the peopled environment, it should recognize and obey the people's social behavior. In this paper, we propose social behavioral robot that stands in line as people do. We have employed the notion of personal space for modeling a line of people and developed stereo vision system to recognize them. We also have developed the mobile robot navigation system that can purchase a cup of coffee even if people are waiting the service in line.

1 INTRODUCTION

Recent research results on mobile robot navigation systems make it promising to utilize them in service fields [7]. In general, the environments where service robots perform tasks are shared with humans. Thus, the robots have to interact with humans, whether they like or not.

Humans also interact with each other. Sometimes, the interactions lead to resource conflicts. In order to maintain order, humans use social rules. For example, at bank counters or grocery stores, human stands in line and wait for his/her turn comes. If a person does not understand nor obey the social rules, he/she will not be able to get the services.

This is also true for the service robots. If a service robot is asked to purchase merchandise and it does not know the social rules of humans, it may keep avoiding the humans who are standing in line as obstacles and will not be able to achieve its task. Therefore, it is also required for service robots to understand the social behaviors of humans, and to obey the rules. But so far, it seems there are no robotic systems that interact with people by taking social behaviors of human into consideration.

There are many aspects to the social rules and behaviors in human society. Among them, standing in line is one of the most highly socialized and crucial skill required for robots which execute tasks in the peopled environments. Therefore, as a first step towards the social behavioral robot, in this research, we will develop the robot which can stand in line with other people.

In the next section, we discuss how the people form line as the result of social behaviors. In section 3, we propose the algorithm for detecting the lined people by vision and the procedures for robot to stand in line. Then we describe the implementation and experimental results in section 4 and 5 respectively.

2 TOWARDS THE ROBOT THAT STANDS IN LINE

In order to realize a robot that stands in line, the robot should be able to recognize a line of people and also should be able to stand in line as people do.

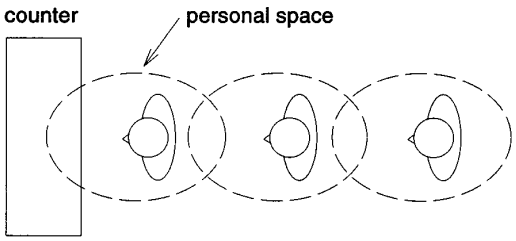

Figure 1: A line of people modeled by a chain of personal spaces.

To recognize the line of people, at first, we should know the social activities of human and how the human form the line. Then we can realize a robot that recognizes social behavior of human and performs actions as people do.

The notion of "human territoriality" or "personal space" has been studied in the research field of cognitive science [3, 6]. Personal space, which is a person's own territory, is oval in shape and is wider towards the front of a person. A person feels uncomfortable when other people are in his/her personal space.

When people form a line, in general, they keep a certain distance from other persons. Also they usually stand so that they faces towards the person in front. We assumed that these phenomenon can be described by borrowing the notion of personal space in follows.

The person who is standing in line keeps enough distance to the person in front so that his/herself may not feels uncomfortable. On the other hand, he/she stands closer enough with the person in front to avoid other person for cutting in the line. Especially by turning the body towards the person in front, the person can express longer range of personal space to other person. Also it increases the possibility of eye contacts which affect other person to feel uncomfortable [6].

0-7803-5731-0/99$10.00 ©1999 IEEE

Thus, we employed this notion and modeled a line of people as a chain of personal spaces shown in figure 1.

Therefore, to recognize a line of people, the robot has to detect each person's position and orientation. In the context of people detection by vision, several face recognition systems have been developed to detect people [1, 5]. Since the robot which is to stand in line observes the lined people from the side or from the back, these methods can not be utilized for our purpose.

Oren et al. have developed people detection system within an image [4]. This system only detects segments of people in an image, but it can not figure out the position and the orientation of the person. So we need to develop our original method for people detection.

3 ROBOT THAT STANDS IN LINE

3.1 Finding Line of People by Vision

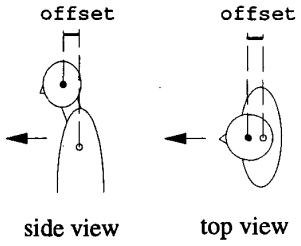

Figure 2: The body model of human.

In order to recognize the position and orientation of people, the three dimensional information is required. In general, stereo vision systems and laser range finders are used for that purpose. Recently the stereo vision system can perform real-time calculation on images of modest size [2]. Also it is cheap in cost compare with laser range finders. So in this research, we use stereo vision system for people detection.

In general, not only human but also obstacles or backgrounds are captured in an image. So we need to discriminate the human data from the disparity images. Also we need to recognize human body orientation as well.

We have modeled the shape of human body as shown in figure 2. We assumed human body is oval shape around the body and is round shape around the head. The neck is the narrowest width in upper body. Also the head juts over the body in front. Based on these assumptions, we have developed people-detection algorithm by vision in follows (see table 1).

At first, 1) capture left and right camera images using two CCD cameras, then 2) calculate disparity image from the two images using triangulation method.

Table 1: The procedure for human detection by vision.

1. capture left and right camera images using two CCD cameras.
2. calculate disparity image from the two images using triangulation method.
3. separate each person's disparity image data from the multiple human's disparity image using a clustering method.
4. find the nose and the chest height position data for each person.
5. find the ellipse that best fits the stereo data at the chest height, and find the circle that best fits the stereo data at the nose height.
6. decide the body's direction by using the center of the ellipse and the circle.

Within the disparity image, multiple persons figures may be captured. At the same time, some noise (such as the object in distant or on wall) may be contained.

To discriminate the disparity data for each person, 3) we use clustering method. The data points which are neighboring within 5cm are connected and categorized as a single cluster so that each person's data to be discriminated. In other words, the chunk of data which are apart more than 5cm are clustered separately. Then we count the number of data points in each cluster and discard the nominal number ones as they are noise or object other than human.

Then we detect the position and orientation of each person from each of the clustered disparity image. For each clustered disparity image, 4) we find the nose and the chest height position data by using the characteristics of human's body shape. At first, we find the highest data point in a cluster and assume it as the top of the head. This is reliable since there are no data above the head in general. Then, we find the narrowest point in width under the head and assume it as the neck position. Finally we estimate the nose height as the middle point between the head top and the neck, and the chest height where 20cm below the neck.

Then 5) we find the ellipse that best fits the stereo data at the chest height, and find the circle that best fits the stereo data at the nose height. The obtained data points (see figure 3-(a)) represent the surface of chest or back at the chest height. It is expected that the data points are forming ellipse shape. Add to that, the data points we can utilize are observed by camera. So we can obtain only the data points that are the half portion of the ellipse.

The mathematical expression of the ellipse which has arbitrary size, location and orientation is as follows.

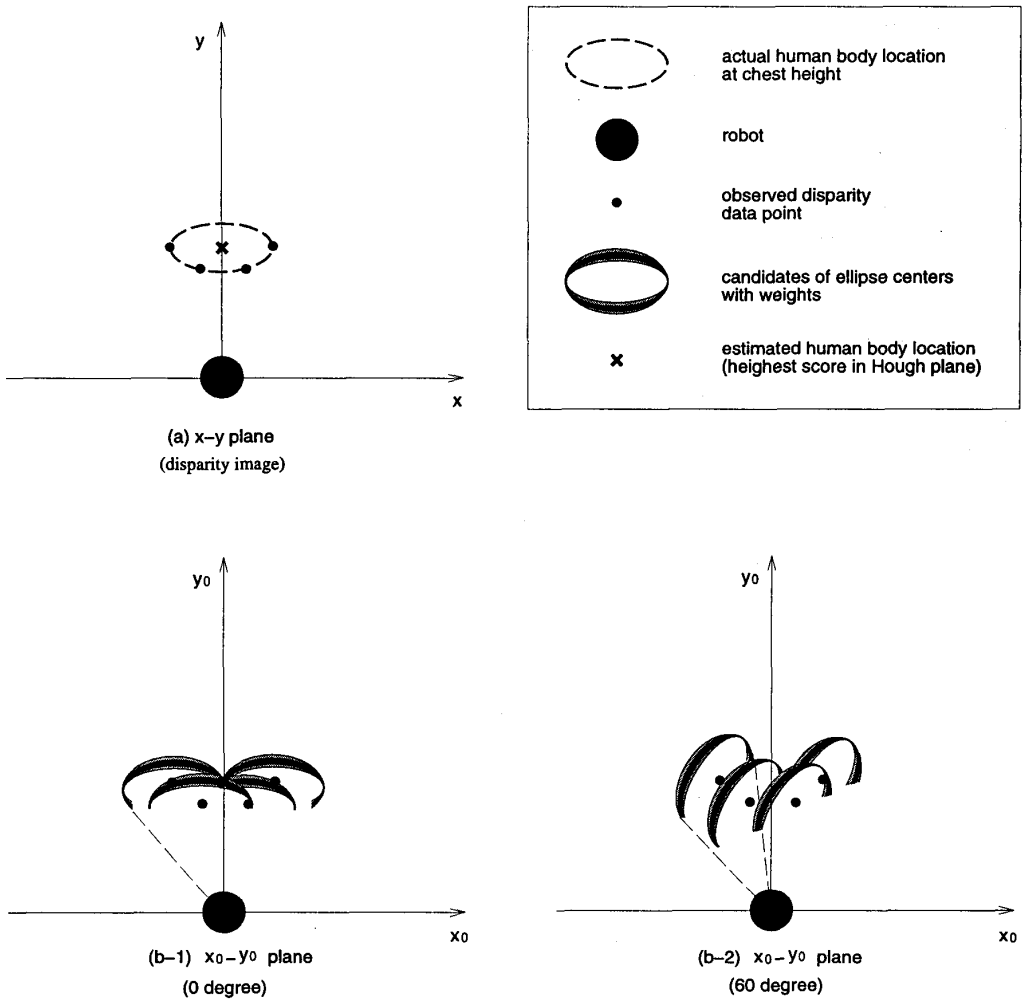

Figure 3: Hough transform used for ellipse fitting.

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a\cos\theta_0\cos\theta + b\sin\theta_0\sin\theta + x_0 \\ -a\sin\theta_0\cos\theta + b\cos\theta_0\sin\theta + y_0 \end{bmatrix} \quad (1)$$

Where, a and b denote the length of major and minor axes, respectively. x_0 and y_0 denote the offset of the center from the origin. θ_0 denotes the inclination of the ellipse. As the sizes of human are almost same, we used constant value for a and b. Thus, the problem is to find the variables x_0, y_0 and θ_0 so that they best fit to the observed data points.

One of the most major algorithms to find parameters in geometrical formula is Hough transform. So we use Hough transform with some modifications to make it optimal for people detection.

Since there are three variables to estimate, we need to have three dimensional Hough transformed space. We discretized θ_0 at 15 degrees and prepared 24 x_0-y_0 planes for each θ_0. For example, Hough images for $\theta_0 = 0$ and $\theta_0 = 60$ are the x_0-y_0 planes as shown in figure 3-(b-1) and (b-2), respectively. The ellipse around each data point denotes the candidates of ellipses in x-y plane. In other words, if you pick up arbitrary point on the ellipse in x_0-y_0 plane and draw an ellipse in x-y plane around the point, that ellipse runs on the original data point.

The procedures of Hough transform to find the best fitting ellipse are as follows. For each data point (x_i, y_i) in x-y plane, draw an ellipse in x_0-y_0 plane as (x_i, y_i) becomes its center. At this time, the number of plots on x_0-y_0 plane are accumulated. We have total of 24 x_0-y_0 planes for different θ_0. So do the same procedures on each plane. Find the most accumulated point among the 24 x_0-y_0 planes, then its (x_0, y_0, θ_0)

is the best fitting ellipse in x-y plane.

In the example shown in figure 3, the data points observed by vision was where the person's orientation is $\theta_0 = 0$. Thus, in the projected x_0-y_0 plane where $\theta_0 = 0$, the plots are accumulated intensively at the position where the person is located. But in the x_0-y_0 plane where $\theta_0 = 60$, the plots are less accumulated.

Add to the above procedures, we have modified original Hough transform as follows. Since noise may be contained in disparity data, if you draw ellipses in x_0-y_0 plane with thin line, the plots may be merely accumulated or casual accumulation at the wrong position may become dominant. So we have distributed the possibility by making the rim of ellipses fat. At the same time, we weighted the rim so that the true ellipse orbit becomes high score.

Also in order to increase the reliability, we employed the heuristics that the candidates of ellipses should be behind the data points. To do so, we used only the portion of the ellipse which can observed by camera as shown in figure 3-(b-2). The ellipse is cut at the tangent lines from the origin (camera position) denoted by dotted lines.

The procedures to find circle at the head position are as same as ellipse fitting. But since the circle doesn't have orientation, we need only one x_0-y_0 plane.

6) Finally, we decide the body direction using the derived center positions of the ellipse and the circle. The result of ellipse fitting on the chest height data contains the ambiguities in the direction whether the person is facing forward or backward. So based on the assumption that the head juts out in front of the body, we resolve the ambiguity by calculating the jutting direction of the head.

3.2 Robot that Stands in Line

Table 2: The procedure to purchase a cup of coffee.

1. get an order from a user.
2. move to the coffee counter.
3. recognize people standing in line.
4. if people are standing in line, find the place to stand and join the line.
5. move up in line by keeping its personal space from a person in front.
6. place an order.
7. recognize a cup which is received.
8. move back to the place where it got the order.
9. hand a cup to the user.

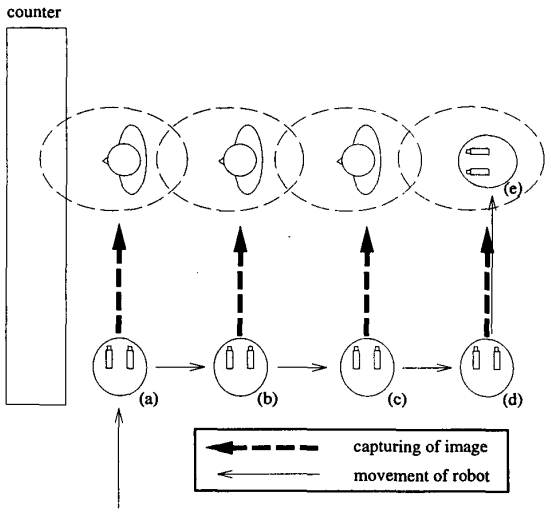

Figure 4: The movement of robot to stand in line.

As the application that requires robot to stand in line, we will develop the service robot to purchase a cup of coffee. The procedure for purchasing the coffee are as follows (see table 2). We have assumed the robot has an environmental map so that it can navigate to the coffee shop and precisely localize in front of the coffee counter.

1) Robot gets an order from user for purchasing coffee by voice, then 2) moves to the coffee shop. The movements of the robot around the coffee counter are depicted as shown in figure 4. The robot moves to position (a) by navigation and precisely localizes it by using the characteristics of the coffee counter such as smooth surfaces and edges.

3) The robot captures image and detects people are standing in line or not by using the algorithms described in 3.1. 4) If there are people standing in line, it moves further from the counter until it finds the place to stand and joins the line (i.e. the movement from position (b) to (e) in figure 4).

Once after the robot joined the line, 5) it moves up in line by keeping its personal space from a person in front so that their personal spaces are connected. The robot should recognize whether the object in front is a person or coffee counter. We utilized the information of laser range finder at the height of human legs. If the observed object surface is smooth enough, it is recognized as coffee counter. If not, it is recognized as human.

At the counter, 6) the robot places an order and 7) waits for the coffee to be handed in the coffee holder of the robot. To recognize if the coffee cup is handed or not, the robot tilts camera so that the cup holder can see and recognizes the availability of cup by vision. Then, 8) the robot moves back to the place where it

got the order. And finally it hands a cup to the user with voice.

4 IMPLEMENTATION

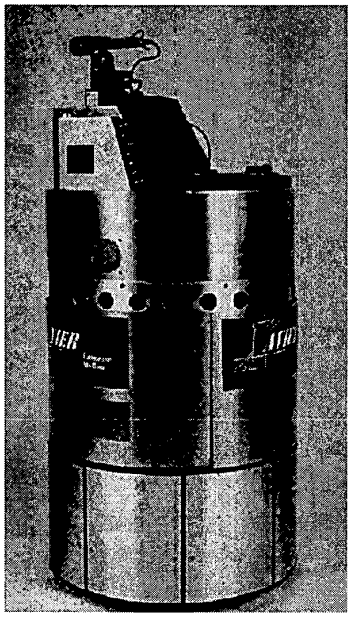

Figure 5: Mobile robot Xavier.

We have implemented above mentioned algorithm on mobile robot Xavier [7] (see figure 5). Xavier is built on top of a 24 inch diameter Real World Interface base. The base is a four-wheeled synchro-drive mechanism that allows for independent control of the translational and rotational velocities. The sensors of Xavier include bump panels, wheel encoders, a 24 element sonar ring, a Nomadics front-pointing laser light striper with a 30 degree field of view, and two Sony monochrome cameras on a Directed Perception pan-tilt head. Xavier also has a speaker and a text-to-speech card. Control, perception, and planning are carried out on two 200 MHz Pentium computers, running Linux.

The software architecture of Xavier is based on Task Control Architecture (TCA) [7] (see figure 6). TCA is constructed by the collection of asynchronous processes and provides facilities for interprocess communication among the processes via central server. So far the following modules have been developed and are ready to utilize for our research.

Base Control Module This module controls the base wheels and provides translation and rotation of the robot. It also manages laser and sonar sensors and reports the current readings to other modules.

Navigation Module This module have environmental map. When the goal position is specified, it plans the path to the goal and navigate the robot to the goal position.

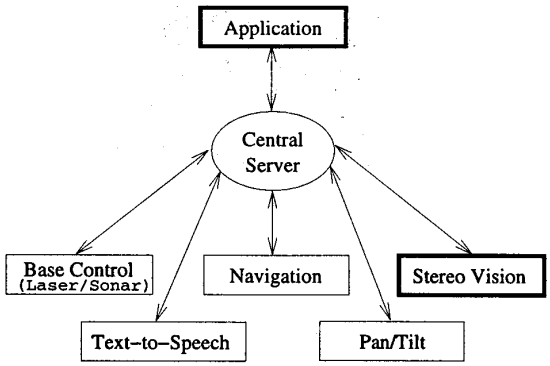

Figure 6: Task Control Architecture of Xavier.

Text-to-Speech Module This module synthesizes and outputs arbitrary natural language text from the speaker.

Pan/Tilt Module This module pans and tilts the cameras.

To realize the robot that stands in line, add to the above mentioned modules, we have developed the stereo vision module and the application module. The stereo vision module utilizes two monochrome CCD cameras and recognizes human based on the algorithm described in section 3.1. For the calculation of disparity image, we utilized SVS[1] stereo engine. The application module controls the whole sequences to purchase coffee as described in section 3.2.

5 EXPERIMENTAL RESULTS

To confirm the efficiency of the system, we have experimented the robot in various situation. Actually we ordered the robot to purchase coffee at the real coffee shop located on the same floor of our laboratory.

An example of captured raw images by stereo cameras and the derived disparity image are as shown in figure 7. The recognized humans with its location and orientations are as shown in figure 8. The sampling time for people detection was about 800 msec.

By experiments we have confirmed that the robot can purchase a cup of coffee reliably even if people are waiting the service in line.

6 CONCLUSION

In this paper, we proposed social behavioral robot that can stand in line as people do. We have developed human detection algorithm based on the notion of per-

[1]SVS is a trade mark of SRI International.

(b) Disparity image

(a-1) Left raw image

(a-2) Right raw image

Figure 7: Raw images and the derived disparity image.

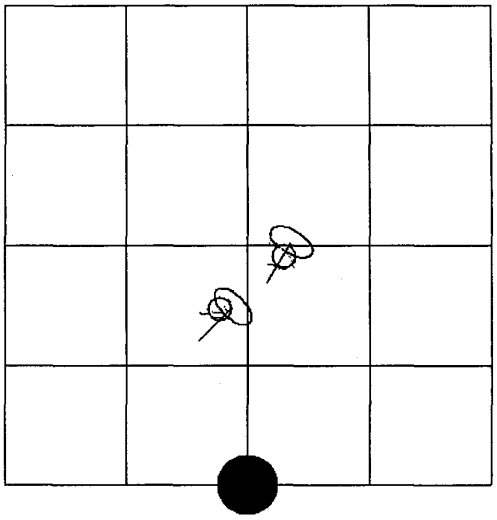

Figure 8: The detected people by the stereo vision system. (black circle: robot, ellipse: chest, circle: head, arrow: body direction)

sonal space. Also we have implemented proposed algorithm on mobile robot Xavier and confirmed its efficiency by experiments.

Currently we are planning to develop other applications that can be modeled by using the notion of personal space (i.e. to join a group of people are standing around talking together).

ACKNOWLEDGMENTS

We would like to thank Kurt Konolige, SRI International, Takeo Kanade and Illah Nourbakhsh, Robotics Institute, CMU for the useful discussions.

REFERENCES

[1] R. Brunelli and T. Poggio, "Template matching: Matched spatial filters and beyond," *MIT AI Lab., A.I. Memo*, (1536), 1995.

[2] K. Konolige, "Small Vision Systems: Hardware and Implementation," Proc. of Eighth International Symposium on Robotics Research, 1997.

[3] M. Malmberg, *Human Territoriality: Survey of behavioural territories in man with preliminary analysis and discussion of meaning*, Mouton Publishers, 1980.

[4] M. Oren, C. Papageorgiou, P. Shinha, E. Osuna, and T. Poggio, "A trainable system for people detection," *Proc. of Image Understanding Workshop*, pp.207–214, 1997.

[5] H. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *In Proc. of IEEE PAMI*, 1998.

[6] R. Sack, *Human Territoriality*, Cambridge University Press, 1986.

[7] R. Simmons, R. Goodwin, K. Zita Haigh, S. Koening and J. O'Sullivan, "A Layered Architecture for Office Delivery Robots," *In Proc. of Autonomous Agents*, pp.245–252, 1997.