CS 4758: You Shall Not Pass! Robotic Sentry with Friend or Foe Identification System

Alessandro Farsi, AEP, 4 PhD, af343
Le Zhang, Independent/Robotics, Senior, lz233
Kenneth Pu, ECE, Senior, kp295

I. Abstract

We propose and implement an application of existing robot learning algorithms to create a robotic sentry: the robot can gauge the hostility of an intruder and make intelligent decisions to aim and shoot.

Using existing machine learning algorithms and stochastic modeling we create ROS code to be utilized on a physical PR2 robot.

II. Introduction

The goal of the project is to develop a robot sentry that guards a specific forbidden zone (e.g. a door or a narrow passage), observing and tracking incoming people (Figure 1). Any approaching human has to stop by the sentry and be acknowledged (standing few seconds facing the robot or performing a specific gesture). If he or she doesn't comply, the sentry will alert and "shoot him down" with Nerf gun triggered by the gripper. Using the Kinect system, we detect the body frames in respect of the sentry and the forbidden zone. Discrimination between safe or trespassing behavior is based on both pose detection and trajectory prediction and is based on supervised learning techniques. Additionally, a Kalman Filter is utilized to predict the steps of the intruder so the robot can aim at a spot in the future to hit the target.

For this project we were originally provided with Willow Garage's PR2 robot and we planned to apply our results to this physical robot. However, due to mechanical failures of the robot we were forced limit our system only to a simulation.

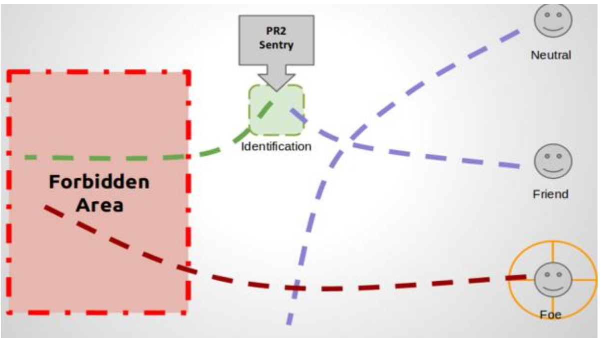

Figure 1. High level description of sentry robot function

III. Related Work

Robotics sentry research has been done by various government and corporations. Papers that deal directly to such applications are not readily available as they pose legitimate security issues. However, of the publicly known sentry-like robots, the most common design is the mounted gun that is human operated and enhanced with sensors help in detecting intruders. For instance, Samsung has implemented the SGR-A1 on the demilitarized zone in South Korea as a physical troop replacement [4].

There is a major leap of faith required for militaries to utilize a fully autonomous robot. The problem not being the lack of performance but rather the reliability of a system that distinguish friend from foe. *Sung et al* were able to implement a hierarchical MEMM with the Microsoft Kinect to distinguish different human activities [1]. In this paper, we simplify the problem of human identification as we are more interested in the integration of a sentry robot that can robustly target and shoot targets. However, we followed the approach of using an RGBD sensor to get more reliable classification results [3].

The main component of our work involves path prediction of sentry targets as well as responsive targeting. In *Hamasaki et al*, the authors approaches this problem by modeling human movement tendencies and creating valid paths for robots to avoid future collisions with humans[4]. We follow a similar approach but instead model our Nerfgun trajectories to enable future collision with human targets. As in *Hamasaki et al*, we make the assumption that human targets do not make sudden changes in movement direction or speed.

IV. Physical Setup

We use Willow Garage PR2 robot [6], with a Kinect depth camera head mounted. Gun is mounted on one arm, firmly attached to the wrist and the gripper.

In order to simplify the process of shooting and reloading. we select Nerfgun Barricade rv-10, a motorized semi-auto revolver with a clip of 10 bullets. Bullets are standard Nerfgun foam bullet with soft tip: once the motor is turned on, triggering requires little pressure.

PR2 gripper are not meant to hold a gun, so we build a frame holding the gun that can be attached to the wrist joint with Velcro strips: the frame is solidly attached and stays in position regardless the joint orientation(fig 2). The gripper can freely move and it is used to partially support the structure weight. An narrow metal extension is attached to one of the gripper ends so that it fits in the trigger. Opening the gripper pushes the trigger and shoot the gun: gripper then needs to close to let the gun reload.

Figure 2. Physical robot connection to PR2 gripper

V. STOCHASTIC NERFGUN BULLET MODELING

In order to create a physical model of the gun, we perform a set of experiments to measure speed and trajectory of the bullets: gun is held in place by a stable grappler(Figure 4) while the entire clip is emptied on a target. Using a camera (23fps) we record and track the bullet flying(Figure 5); tracking is done using the freeware software Tracking. In a first configuration of the experiment, a target is positioned at a variable distance, with the camera is facing it, so that we can measure where the bullet lands on the target; in a second configuration, the camera is placed perpendicular to shooting direction, so that the entire trajectory can be recorded.

We shoot 50 bullets to targets placed at distances varying from 2.8 m to 5.7 m for a total of 80 bullets. We also record 20 trajectories in the second configuration. Analyzing the data, we observe that the bullets leaves the gun with an initial momentum and flies under the only effect of gravity (air friction is not relevant in our condition). From the spreading of the landing points we extract a radial distribution of the bullet direction.

A model the bullet as a frictionless mass with initial constant momentum (v = 17.8 m/s) and a direction normally distributed around the aiming direction (delta theta = 1 degree, extracted from data) fits well the data.

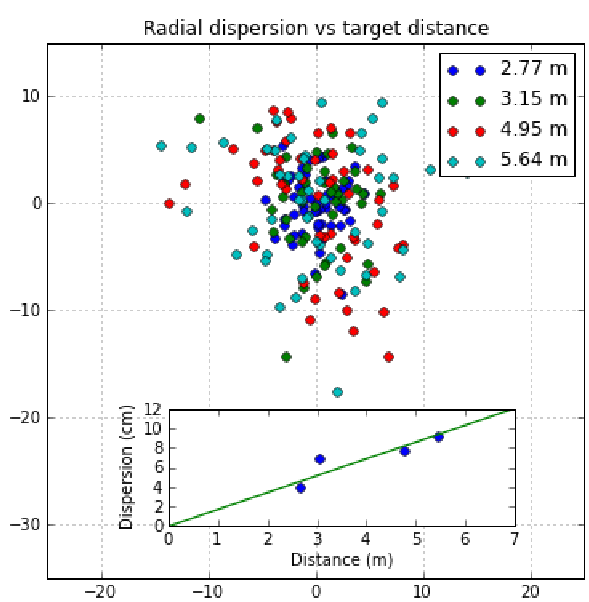

Figure 3. Radial dispersion vs target distance

Figure 4. Setup to test nerfgun shot distribution

Figure 5. Determining bullet speed with software

VI. AIMING AND INVERSE KINEMATICS

A sentry has to be fast in aiming and shooting a moving target, so we simplify the inverse kinematics problem (i.e. transforming the gun orientation to robot arm joints angles) by placing the robot on a “crane” configuration (Figure 6) that directly maps joints angles to gun yaw and pitch.

The frame transformation between robot wrist and gun is hardcoded in the aiming routine.

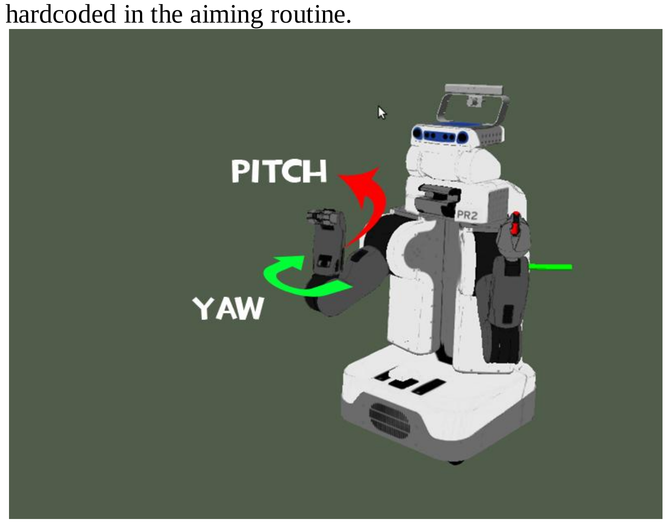

Figure 6. Configuration of PR2 robot when shooting

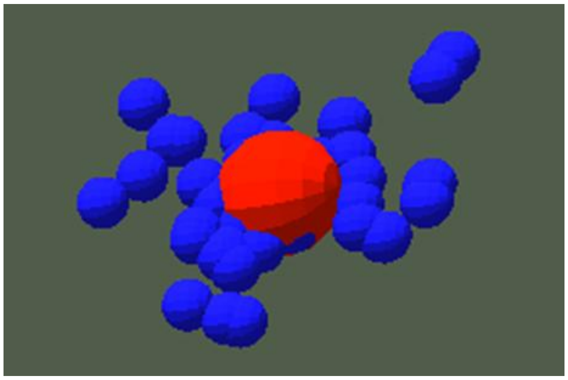

Figure 7. Stochastic distribution of bullet (in blue) on target (red)

VII. THREAT DETECTION AND TRACKING

A. Hostility Classification Using SVM

Another important feature of a robotic sentry is the ability to reliably distinguish between friendly and hostile targets. In order to provide this classification functionality, we first come up with a list of classes of behaviors that we would deem to be either benign or suspicious, which can be seen below in Figure 8.

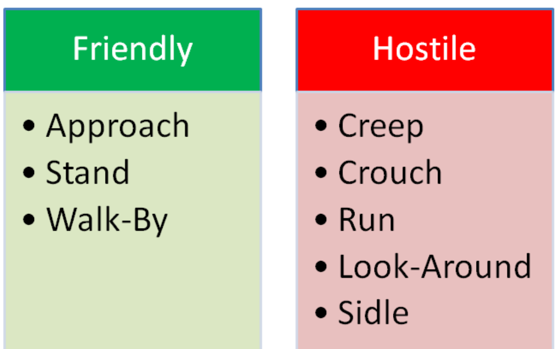

Figure 8. Diffent types of friendly and hostile behaviors

Next, using a script we previously wrote allowing us to record joint position and orientation data, we generated two datasets for each group member performing each class of behavior(Figures 9-10) in front of the Kinect, for a total of 48 sets. Each dataset contains position and orientation data of the subject's joints at incremental time-steps as they go through a certain set of motions. Of these datasets, 8 sets (one for each class) were set aside for testing purposes, resulting in 40 training sets and 8 testing sets. After normalizing the data so that all joint positions are in relation to the torso, we use SVM to build a model from our training sets.

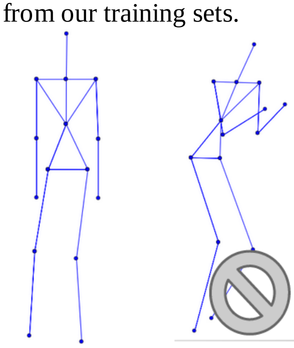

Figure 9. Friendly vs hostile skeleton

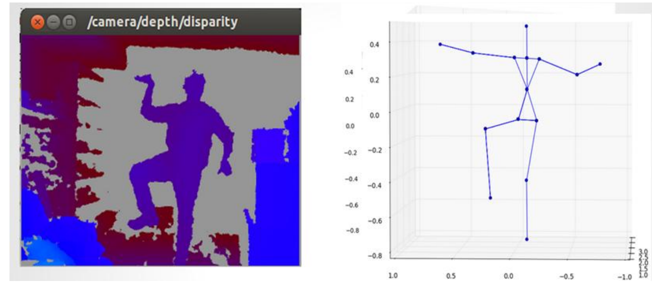

Figure 10. Visualizing skeletons with Kinect

Our results are as shown in Table 1, where we attempt to gauge the accuracy of our method using varying sized feature vectors. The feature vectors are composed of frame information from n time steps.

TABLE I. CLASSIFICATION ACCURACY VS VECTOR LENGTH

Vector Length (n-time steps)	*Accuracy*
1	78%
2	75%
3	71%
4	71%
5	71%

It seems that using the single time-step as the feature vector gives us the best results. The accuracy plateaus past using 3 time-steps as a single vector. Additionally, there is the likelihood that the system will quickly jump from friendly to hostile depending on a single frame of information. To give the robot more confidence in its behavior classification, we look at the classification results of multiple frames and sum the confidence value of each frame before we make a decision. We utilize a state machine to make it difficult for the robot to make sudden transitions between hostile and friendly behavior. In Table 2 below we show that this method of summing SVM certainty over time works well in improving our accuracy.

TABLE II. CLASSIFICATION ACCURACY VS NUMBER OF FRAMES USED

Number of Frames	*Accuracy*
2	80.55%
3	81.77%
4	83.33%
5	84.35%
6	84.35%

There is an upper limit to the number of frames we can apply in this situation. Too many frames would result in having a system that takes too long to judge hostility. Additionally, the behavior of the target n seconds in the past might no longer be relevant to the robot. For our ROS application we used 10 frames.

B. Position Tracking and Prediction Using Kalman Filter

Using the Kinect and the ROS API, we were able to track the positions and orientations of each joint of a person moving in front of the Kinect. We developed a script to continuously log this joint information to a text file, allowing us to record a person's movements at a rate of 10 Hz. With this program we were able to generate several training sets of both friendly and hostile behavior.

In addition, with slight modifications to our script we were able to track the x,y coordinates of a person's torso to get a plot of the person's location over time (in relation to the Kinect). Once again, with the script we were able to display either one or multiple paths as seen in Figure 11.

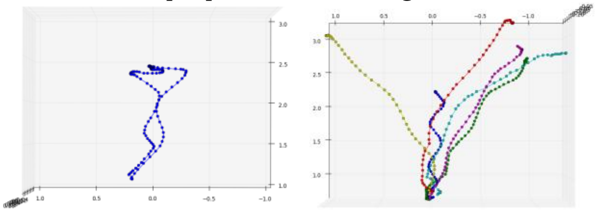

Figure 11. Position tracking in time

By feeding this trajectory information into a Kalman filter, we were also able to lay the groundwork for basic path prediction. The current implementation simply takes the person's velocity at a certain frame and uses it to predict the person's position at the next step.

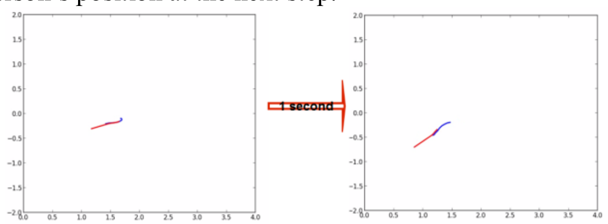

Figure 12. Kalman filter path prediction: path (blue) vs prediction (red)

We tweaked the various parameters(noise, observation, state matrix) for the Kalman filter through live trials with the Kinect and observed acceptable performance. As seen in the figure above (Figure 12) we predict smooth trajectories that matches the path history 1 second prior. We were able to test this function live with a simulator PR2(Figure 13).

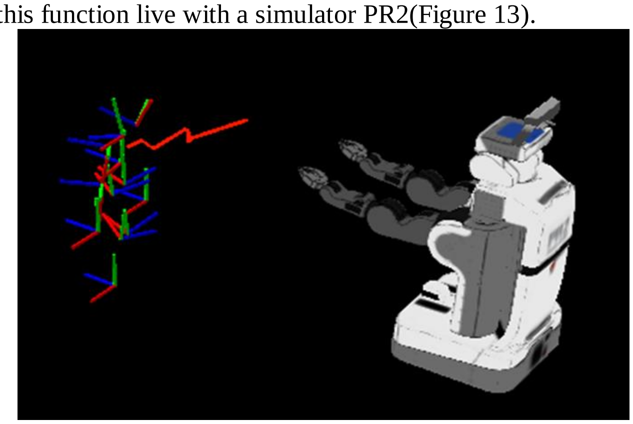

Figure 13. Live trajectory generation in simulation with Kinect input

C. *Optimal Bullet Trajectory*

Given a certain trajectory (time + space) of the target, the sentry needs to decide when and where to shoot the bullet. Indeed the bullets trajectory has to intersect the target one at the right time, but we need maximize the probabilities of hitting the target, given the uncertainties of the aim.

We utilize the following planning algorithm: first, we sample the target predicted trajectory $\{S_t\}$, and - neglecting the angular uncertainty on the bullet momentum - we calculate orientation and time of flight of a rough aim by inverting the Newtonian kinematics of the projectile.

As a second step, we use a Montecarlo method to estimate the probability of hitting any other point of the target trajectory: for each rough aim needed to hit a certain $S_{t\mathrm{i}}$, we sample the random distribution of the bullet $\{S_b(t)\}$ at different time t, and using a heuristical probability of hitting the target as

$$\begin{cases} P = 1 & if\ \text{Distance}(S_t, S_b) < r = 0.3m \text{ (radius target)} \\ P = \dfrac{1}{\text{Distance}(S_t, S_b)} & otehwise \end{cases}$$

we obtain the total probability of hitting the target in any other instant of its trajectory.

Doing so, shooting is preferred when the target is moving parallel to aim rather than perpendicular.

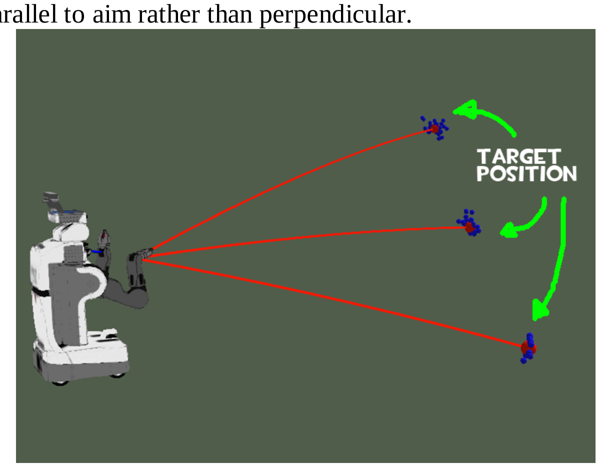

Figure 14. For each sample target position $\{S_t\}$, the system calculates a rough aim inverting the Newtonian kinematics of the bullet (red line)

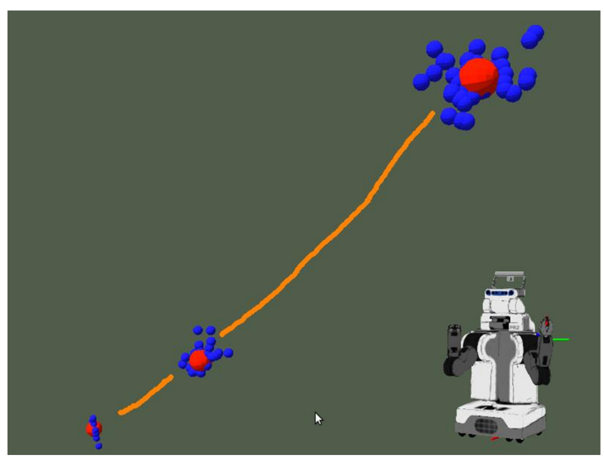

Figure 15. For each rough aim, the system samples where the bullet would land (blue dots) and compares it with the trajectory of the target (red dots)

VIII. Sentry Main Program

A. *ROS*

We utilize WillowGarage ROS library to interface with PR2: ROS provides foundations for the creation of software to control the robot. All our code is written in Python, but takes advantage of compiled fast numerical libraries such as Numpy, PyKdl and SVM in addition to ROS codebase and libraries such as OpenNi, ForwardKinematics and ArmMovement.

Visualization of the various components of the project is done in RViz, while the simulation of robot physics is performed in Gazebo.

B. *Overview of the System*

'mastermind.py' is our main node script that controls the final logic of the robots and calls the different components of the project. It is structured as a state machine, constantly tracking and analyzing skeleton tracking stream. Once a target is flagged as 'foe', the aiming and shooting routine is called. The code follows as simple state machine as shown in Figure 16.

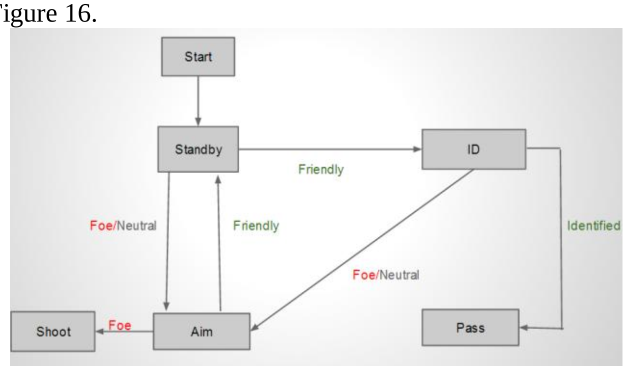

Figure 16. State machine used by mastermind.py

IX. OBSERVATIONS

In our project, we developed a system to turn a PR2 robot into an automatic sentry: we identify the major components (target tracking, behavior identification, aiming prediction and optimization) and we implemented them using standard machine learning techniques. In addition, we explored ROS libraries and taken advantage of the functionalities that it provides.

For our project we were conditioned by two elements: the first is the fact we were using a real world complex object such as the Nerfgun, that is not easily manipulated by the robot gripper. Creating a custom frame for attaching it to the robot we removed the need special manipulation, simplifying the codebase.

The second aspect is the real time response by the robot: in order to track a shoot a moving target, the robot must be able to plan and move fast. For this reason we did not rely on the standard PR2 motion planning, but we implemented a simplified version that can act much faster (but operates in more special and controlled case).

X. FUTURE WORK

In future research we would like to apply our software to the physical PR2 as there are many problems that cannot be solved in simulation. We have solved the major physical interfacing issue but the responsiveness of the shoot and aim sequence will require testing on the PR2.

Additionally it would be interesting to pursue other algorithms to increase the reliability of our hostility detection system. It is arguable that for autonomous weapons, hostility detection is much more important than the performance of the weapon. A potential solution is to additionally use computer vision to use specific markers(hats, uniforms, etc) on a target as a secondary basis for hostility. Having multiple method to determine hostility will significantly increase the robot's confidence in hostility detection.

REFERENCES

[1] Jae Y. Sung, Colin Ponce, Bart Selman, Ashutosh Saxena. Human Activity Detection from RGBD Images, In AAAI workshop on Pattern, Activity and Intent Recognition (PAIR), 2011.

[2] Shue, S.; Hargrove, C.; Conrad, J., "Low cost semi-autonomous sentry robot," Southeastcon, 2012 Proceedings of IEEE , vol., no., pp.1,5, 15-18 March 2012 URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6196937&isnumber=6196883

[3] Zainordin, Faeznor Diana; Hwea Yee Lee; Sani, Noor Atikah; Yong Min Wong; Chee Seng Chan, "Human pose recognition using Kinect and rule-based system," World Automation Congress (WAC), 2012 , vol., no., pp.1,6, 24-28 June 2012 URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6321032&isnumber=6320876

[4] Kumagai, J. "A Robotic Sentry for Korea's Demilitarized Zone", MSPEC, 2007 Proceedings of IEEE, pp. 16,17, March 2007 URL:https://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=4119213

[5] Hamasaki, S. Tamura, Y. ; Yamashita, A. ; Asama, H. "Prediction of human's movement for collision avoidance of mobile robot", "Robotics and Biomimetics (ROBIO), 2011 IEEE International Conference on", pp. 1633,1638, 7-11 Dec. 2011 URL:https://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=4119213

[6] Willow Garage PR2 Overview webpage http://www.willowgarage.com/pages/pr2/overview