Online-learning and Attention-based Obstacle Avoidance Using a Range Finder

Shuqing Zeng and Juyang Weng
Department of Computer Science and Engineering
Michigan State University
East Lansing, MI 48824-1226, USA
{zengshuq, weng}@cse.msu.edu

Abstract

We considered the problem of developing local reactive obstacle-avoidance behaviors by a mobile robot through online real-time learning. The robot operated in an unknown bounded 2-D environment populated by static or moving obstacles (with slow speeds) of arbitrary shape. The sensory perception was based on a laser range finder. We presented a learning-based approach to the problem. To greatly reduce the number of training samples needed, an attentional mechanism was used. An efficient, real-time implementation of the approach had been tested, demonstrating smooth obstacle-avoidance behaviors in a corridor with a crowd of moving students as well as static obstacles.

Introduction

The real-time range-based obstacle avoidance in dynamic environments has been studied by many researchers. The local reactive approaches are usually used in unknown or partially unknown environments since they reduce the problem's complexity by computing short-term actions based on current local sensing. For example, the curvature-velocity method (Simmons 1996) and the dynamic window (DW) approach (Fox, Burgard, & Thrun 1997) formulate obstacle avoidance as a constrained optimization problem in a 2-D velocity space. They assume that the robot moves in circular paths. Obstacles and the robot's dynamics are considered by restricting the search space to a set of admissible velocities.

In contrast with the above efforts that concentrate on action generation without requiring sophisticated perception, a series of research deals with perception-guided behaviors. An important direction of research, the appearance-based method (Pomerleau 1991) (Hwang & Weng 1997) (Chen & Weng 2000), aims at reducing or avoiding those human-defined features for better adaptation of unknown scenes. The need to process high dimensional sensory vector inputs in appearance-based methods brings out a sharp difference between behavior modeling and perceptual modeling: the effectors of a robot are known with the former, but the sensory space is extremely complex and unknown with the latter and, therefore, very challenging.

In this paper, we present an approach of developing a local obstacle avoidance behavior by a mobile humanoid through online real-time incremental learning. A major distinction of the approach is that we used the appearance-based approach for range-map learning, rather than an environment-dependent algorithm (e.g., obstacle segmentation and classification) for obstacle avoidance. The new appearance-based learning method was able to distinguish small range map differences that are critical in altering the navigation behavior (e.g., passable and not passable sections). In principle, the appearance-based method is complete in the sense that it is able to learn any complex function that maps from the range-map space to the behavior space. This also implies that the number of training samples that are required to approximate the complex function is very large. To reduce the number of training samples required, we introduced the attentional mechanism which dynamically selected regions in near approximity for analysis and treated other regions as negligible for the purpose of local object avoidance. To indicate the benefit of using the attentional mechanism, we analytically derived a lower bound of the sample reduction ratio. Further, online training was used so that the trainer could dynamically choose the training scenarios according to the system's current strengths and weaknesses, further reducing the time and samples of training.

Approach

Problem statement

We consider the obstacle avoidance behavior as a reactive decision process, which converts the current range map to action. The robot does not sense and store scene configuration (e.g., global map of the environment) nor the global robot position. That is, we assume that the current range map contains all of the information that is sufficient for robots to derive the next motor control signal. In fact, it uses the real world as its major representation. In the work presented here, the robot's only goal is to move safely according to the scene: It has no target location. Such a navigation system is useful for applications where a human guides global motion but local motion is autonomous.

The range scanner observes $\mathbf{r}(t) \in \mathcal{R} \subset R^n$ at time t, where $\mathcal{R}$ denotes the space of all possible range images in a specific environment. $\mathbf{r}(t)$ is a vector of distance, whose ith

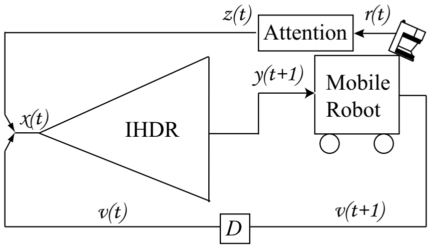

Figure 1: The control architecture of the range-based navigation. “Attention” denotes an attentional module. $\mathbf{r}_p(t)$ denotes the retrieved range prototype.

component r_i denotes the distance to the nearest obstacle at a specific angle. The vector $\mathbf{v}(t)$ gives the current velocities of the vehicle at time t, which are measured by the vehicle’s encoders. The vectors $\mathbf{r}(t)$ and $\mathbf{v}(t)$ are given by two sensors whose dimension and scale are different, thus, we need a normalization procedure when merging the two vectors together as the integrated observation vector.

Definition 1 *The vector* $\mathbf{x}(t)$ *denotes the system’s observation of the environment at time* t. *It is defined as:*

$$\mathbf{x}(t) = (\frac{\mathbf{r}(t) - \bar{\mathbf{r}}}{w_r}, \frac{\mathbf{v}(t) - \bar{\mathbf{v}}}{w_v}) \in \mathcal{X}, \qquad (1)$$

where w_r *and* w_v *are two positive numbers that denote the scatter measurements of the variates* $\mathbf{r}$ *and* $\mathbf{v}$, *respectively; and* $\bar{\mathbf{r}}$ *and* $\bar{\mathbf{v}}$ *are sample means.*

The action vector $\mathbf{y}(t) \in \mathcal{Y}$ consists of control signals sent to all of the effectors at time t, where $\mathcal{Y}$ denotes the sample space action vectors.

The obstacle avoidance behavior can be formulated as a mapping $f : \mathcal{X} \mapsto \mathcal{Y}$, i.e., the primed action $\mathbf{y}(t+1)$ (the signal sent to the motors) is a function of $\mathbf{x}(t)$:

$$\mathbf{y}(t+1) = f(\mathbf{x}(t)). \qquad (2)$$

Fig. 1 shows the coarse control architecture of the presented approach. An Incremental Hierarchical Discriminating Regression (IHDR) (Hwang & Weng 2000) tree is generated to estimate the control signal $\mathbf{y}$ from $\mathbf{x}$. The current input range image $\mathbf{r}(t)$ and the vehicle’s velocities $\mathbf{v}(t)$ are used for deriving the next control signal $\mathbf{y}(t+1)$. An attentional module is added to extract partial views from a whole view of a scan.

Attentional mechanism

Direct use an image as a long vector for statistical feature derivation and learning is called the appearance-based approach in the computer vision community. Usually the appearance-based approach uses monolithic views where the entire range data (or visual image) frame is treated as a single entity. However, the importance of signal components is not uniform. There are cases where appearances of two scenes are quite similar globally, but different actions are required. Further, similar actions are needed where the appearance of two scenes look quite different globally. Both cases indicate that there are critical areas where differences critically determine the action needed. This necessitates an attentional mechanism to select such critical areas.

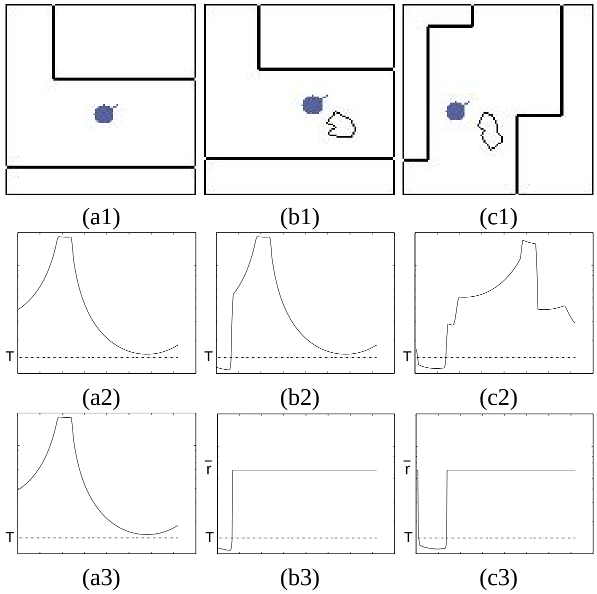

Figure 2: (a1), (b1) and (c1) show three scenarios along the robot’s path. The solid small circles denote the robot with a short line segment at the front indicating orientation. Thick lines mark the walls along the corridor. T and $\bar{r}$ denote the threshold and the mean, respectively. (a2), (b2) and (c2) are range images taken by the robot at the three scenarios, respectively. (a3), (b3) and (c3) are the corresponding images after passing the attentional module. The diagrams in lower two rows use logarithmic scales for the Y-axis. We can see that the distance between (a3) and (b3) becomes larger while the distance between (b3) and (c3) becomes smaller.

For example, in Fig. 2, (a1), (b1) and (c1) show three scenarios along the robot’s path. Range images (a2) and (b2) are quite similar globally judged from the entire image (except the critical area on the left side). In the context of an appearance-based method, this means that the distance (e.g., Euclidean one) between the two is small. They require different actions: turning right for (a1), going straight or turning left for (b1). In another case, range images (b2) and (c2) are very different globally, but their desired actions are similar: going straight or turning left. Thus, it is difficult to discriminate the three cases correctly by using a distance metric defined on the entire image. But, if we look at the left subregions in (a2), (b2) and (c2) of Fig. 2, we can see that the similarities and differences are clear. Without a capability to attend to this critical region, the learning system requires significantly more training samples when complex scenarios are considered.

In the above three cases, the critical area is the input component where range readings are very small. This is true, in general, because near obstacles determine heading more than, and often take precedence over, far-away objects. As we shall see later, this can be accomplished by an attentional mechanism.

We first define the scalar attentional effector.

Definition 2 *The operation of the attentional effector* $a(t)$ *for input* $r(t)$ *and output* $z(t)$ *is defined by:*

$$z(t) = g(r(t), a(t)) = \begin{cases} r(t) & a(t) = 1, \\ \bar{r} & a(t) = 0, \end{cases} \qquad (3)$$

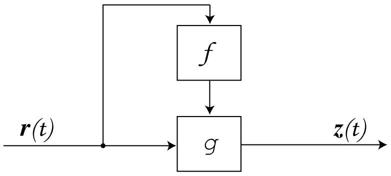

Figure 3: The attentional signal generator f and attentional executor g. $\mathbf{r}(t)$ and $\mathbf{z}(t)$ denote the input and output, respectively.

where $\bar{r}$ denotes the sample mean of the raw signal $r(t)$.

For intended application, we would like to have $a(t)$ to behave in the following way. First, when all input components have large values, the attentional selection is in its default mode, turning on all components. Second, when there are nearby objects, the attentional selection activates only nearby objects which are critical for object avoidance while far-away objects are replaced by their mean readings. This attentional action can be realized by two programmed functions g and f:

$$z_i(t) = g(r_i(t), a_i(t)) \tag{4}$$

and

$$a_i(t) = f(\mathbf{r}(t)) = \begin{cases} 1 & \text{if } r_i < T \text{ or } \forall j \;\; r_j(t) \geq T, \\ 0 & \text{otherwise,} \end{cases} \tag{5}$$

where T is a threshold, $i = 1, 2, ..., n$, and $\mathbf{r}(t) = (r_1(t), r_2(t), ..., r_n(t))$ denotes the input vector. We write $\mathbf{z}(t) = (z_1(t), z_2(t), ..., z_n(t))$ as the output of the attentional action and $\mathbf{a}(t) = (a_1(t), a_2(t), ..., a_n(t))$ as the attention vector. The above function f will suppress some far-away components ($a_i(t) = 0$) if there are objects closer than T. If all readings are far-away, we do not want to turn the attention off completely and, therefore, we leave all attentional effectors on ($\forall j, \;\; a_j(t) = 1$). This operation is illustrated in Fig. 3.

In practice, this raw attentional vector $\mathbf{a}(t)$ is smoothed by convoluting with a flat window, as

$$a'_i(t) = [\frac{1}{11} \sum_{j=i-5}^{i+5} a_j(t)],$$

where $[\cdot]$ denotes rounding to the nearest integer. This smoothing serves to eliminate point-wise noise and to provide a neighborhood influence to the output attentional vector.

In Fig. 2, the readings of the left part of diagrams (b2) and (c2) are smaller than T. Thus, only the left part passes through the attentional mechanism without change whereas other parts are suppressed by being set to the mean, as shown in (b3) and (c3) of Fig. 2. This is needed for the robot to pass through tight areas, where a small change in the width of a gap determines whether the robot can pass. The attentional mechanism enables the robot to focus on critical areas (i.e., parts with close range) and, thus, the learned behaviors sensitively depend on the attended part of the range map. All range readings are attended when there is no nearby object as shown by (a2) and (a3) of Fig. 2.

In Fig. 1, the learner IHDR is a hierarchically organized high-dimensional regression algorithm. In order to develop stable collision-avoidance behaviors, the robot needs sufficient training samples. Here, we show that the attentional mechanism greatly reduces the number of necessary training samples when there are objects close to the robot. To quantitatively analyze the attentional mechanism proposed, we make the following definition.

Table 1: The lower bound of the reduction ratio at several parametric settings.

Parameters	Reduced size
$r_m = 50$m, $\Delta = 0.2$m, $T = 2$m, and $h = 10$	9.51×10^{20}
$r_m = 10$m, $\Delta = 0.2$m, $T = 2$m, and $h = 15$	7.52×10^{22}

Definition 3 *Consider a scanner operating in a environment, which can be approximated by piecewise 3-D planes (for simplicity analysis only). Each range map* **r** *can be approximated by a polygon with h segments (as in Fig. 4 (a)). $P = (p_1, p_2, ..., p_h)$ is the map, where the ith end point p_i is denoted by its polar coordinate (r_i, α_i). Without lost of generality, the angle coordinate are sorted: $\alpha_1 < \alpha_2 < \cdots < \alpha_h$. $P' = (p'_1, p'_2, ..., p'_h)$ is the post-attentional map, where $p'_i = (z_i, \alpha_i)$ whose range z_i has been defined earlier.*

Remark. The larger h is, the closer the approximation of **r** in general. In a particular case, the polygon representation becomes a regular grid when $h = n$.

We write the post-attentional approximation P' as the function of P, i.e.,

$$P' = g^*(P). \tag{6}$$

The attentional mechanism, defined in Eq. (6), is not a one-to-one mapping, as shown in Fig. 4. The post-attentional map P' is the representative for a set of pre-attentional maps besides P if condition C: $\exists p_j \; p_j \in P \wedge l_j < T$ is satisfied. We denote this set by $\mathcal{R}(P')$, i.e. $\mathcal{R}(P') \equiv \{P | g^*(P) = P'\}$. The following theorem gives a lower bound of the average size of $\mathcal{R}(P')$ when there are objects within the distance T of the robot.

Theorem 1 *Let Δ and r_m denote, respectively, the range resolution and maximum distance of each radial line. If the ith end point p_i's radial length r_i is a random variable with a uniform distribution in the sample space $\{0, \Delta, 2\Delta, ..., r_m\}$. Then the average size of the set $\mathcal{R}(P')$ conditioned on C is:*

$$E_{P'}\{\mathcal{R}(P') | C\} > \frac{(1-p)hq^{h-1}}{1-p^h}, \tag{7}$$

where $q = (r_m - T)/\Delta$, $p = (r_m - T)/r_m$, and $E_{P'}\{\cdot | C\}$ denotes the expectation on condition C.

We relegate the proof of Theorem 1 to Appendix A. Table. 1 shows the lower bound of the size due to attention at two typical parametric settings. We see that the reduction is large. Of course the size of remaining space to learn is also large, the ratio of space reduced over original space is roughly p.

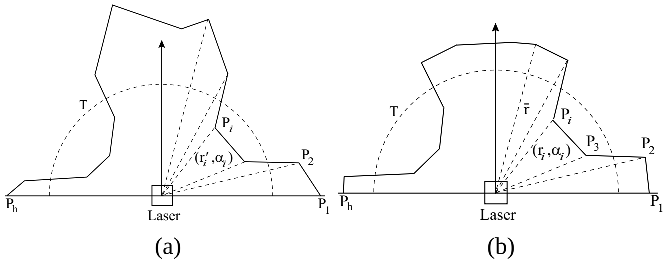

Figure 4: (a) shows that the pre-attentional map **r** is approximated by a polygon P. P is specified by h end points: $p_1, p_2, ..., p_h$. The kth end point is denoted by (l_k, α_k). (b) shows the post-attentional approximation P', after the attentional function g^*. Those end-points whose distances are larger than T are set to $\bar{r}$. The half-circle with symbol T shows the area the robot pays attention to. We can see numerous pre-attentional approximations map to a single post-attentional approximation P'. It is clear that the points outside the half-circle of (a) have the freedom to change positions without affecting the shape of (b).

IHDR: Associative memory mapping engine

The IHDR realizes a regression $f : \mathcal{X} \mapsto \mathcal{Y}$ through context-addressable recall for each high-dimensional input vector $\mathbf{x}(t)$. IHDR is not new for this paper, for details see (Hwang & Weng 2000). Its function is similar to CART, C5.0 and SVM, but performs significantly better for high-dimensional input.

IHDR dynamically approximates the input and output spaces using a large but finite number of clusters represented as high dimensional vectors. The X-clusters in the $\mathcal{X}$ space represent a larger number of context prototypes. IHDR automatically derives discriminating feature subspaces in a coarse-to-fine manner from input space $\mathcal{X}$ in order to generate a tree architecture of memory self-organization. Shown in Fig. 5, a linear subspace is spanned by automatically derived discriminating features at the root of the tree. The features are most discriminative in the sense they span the linear subspace that passes the space of between-class scatter. In this way input components that are irrelevant to the mapping's output are disregarded to achieve better discrimination and generalization.

A probability-based nonlinear partition in the subspace divides the entire input space $\mathcal{X}$ into a number of regions, as shown in Fig. 6. Such a coarse-to-fine partition of input space is carried out recursively until the node has received only a few samples of prototypes corresponding to the most detailed cases. Each prototype is then associated with the desired outputs in the $\mathcal{Y}$ space. The tree structure recursively excludes many far-away prototypes from consideration (e.g., a near object does not search far-away objects), thus the time to retrieve the associated contexts to update the mapping for each input x is $O(\log(n))$, where n is the number of leaf nodes. This extremely low time complexity is essential for real-time online learning with a very large memory.

The robotic system and online training procedure

The tests were performed in a humanoid robot, called Dav, built in the Embodied Intelligence Laboratory at Michigan State University. Details about the Dav robot, please refer to (Han *et al.* 2002) and (Zeng, Cherba, & Weng 2003).

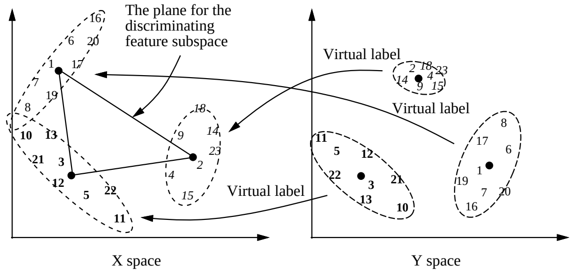

Figure 5: Y-clusters in space $\mathcal{Y}$ and the corresponding X-clusters in space $\mathcal{X}$. Each sample is indicated by a number which denotes the order of arrival. The first and second order statistics are updated for each cluster.

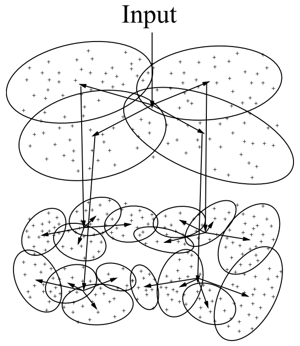

Figure 6: The incrementally generated IHDR tree. Each node has its own most discriminating feature (MDF) subspace $\mathcal{D}$ which partitions its own input samples into its children.

Mounted on the front of Dav, the laser scanner (SICK PLS) tilts down 3.8° for possible low objects. The local vehicle coordinate system and control variables are depicted in Fig. 7. During training, the control variables (θ, d) are given interactively by the position of the mouse pointer P through a GUI interface. Once the trainer clicks the mouse button, the following equations are used to compute the imposed (taught) action $\mathbf{y} = (v, \omega)$:

$$\omega = -K_p(\pi/2 - \theta)$$
$$v = \begin{cases} K_v d & P \in \text{Area I}, \\ 0 & P \in \text{Area II}, \end{cases} \quad (8)$$

where K_p and K_v are two predetermined positive constants. Area II corresponds to rotation about the center of the robot with $v = 0$.

Dav's drive-base has four wheels, each driven by two DC motors. Let $\dot{\mathbf{q}}$ denote the velocity readings of the encoders of four wheels. Suppose v_x and v_y denote the base's translation velocities, and ω denotes the angular velocity of the base. By assuming that the wheels do not slip, the kinematics of the base is (Zeng, Cherba, & Weng 2003):

$$\dot{\mathbf{q}} = B(v_x, v_y, \omega)^T, \quad (9)$$

where B is an 8 × 3 matrix, decided by the wheels' configuration (known). The base velocities $(v_x, v_y, \omega)^T$ is not directly available to learning. It can be estimated from the

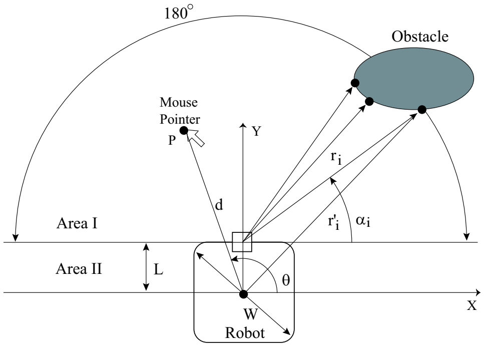

Figure 7: The local coordinate system and control variables. Once the mouse buttons are clicked, the position of the mouse pointer (θ, d) gives the imposed action, where θ denotes the desired steering angles, and d controls the speed of the base.

wheels' speed vector $\dot{\mathbf{q}}$ in a least-square-error sense:

$$\mathbf{v} = (v_x, v_y, \omega)^T = (B^T B)^{-1} B^T \dot{\mathbf{q}}. \qquad (10)$$

In this paper, we use two velocities, (v_y, ω), as the control vector $\mathbf{y}$. Thus, the IHDR tree learns the following mapping incrementally:

$$\mathbf{y}(t+1) = f(\mathbf{z}(t), \mathbf{v}(t)).$$

During interactive learning, $\mathbf{y}$ is given. Whenever $\mathbf{y}$ is not given, IHDR approximates f while it performs (testing). At the low level, the controller servoes $\dot{\mathbf{q}}$ based on $\mathbf{y}$.

Online incremental training

The learning algorithm is outlined as follows:

1. At time frame t, grab a new laser map $\mathbf{r}(t)$ and the wheels' velocity $\dot{\mathbf{q}}(t)$. Use Eq. (10) to calculate the base's velocity $\mathbf{v}(t)$.

2. Computer $\mathbf{a}(t)$ based on $\mathbf{r}(t)$ using Eq. (5). Apply attention $\mathbf{a}(t)$ to given $\mathbf{z}(t)$ using Eq. (4). Merge $\mathbf{z}(t)$ and the current vehicle's velocities, $\mathbf{v}(t)$, into a single vector $\mathbf{x}(t)$ using Eq. (1).

3. If the mouse button is clicked (training), Eq. (8) is used to calculate the imposed action $\mathbf{y}(t)$, then go to step 4. Otherwise go to step 6.

4. Use input-output pair $(\mathbf{x}(t), \mathbf{y}(t))$ to train the IHDR tree as one incremental step.

5. Send the action $\mathbf{y}(t)$ to the controller which gives $\dot{\mathbf{q}}(t+1)$. Increment t by 1 and go to step 1.

6. Query the IHDR tree by calling the retrieval procedure and get the primed action $\mathbf{y}(t+1)$. Send $\mathbf{y}(t+1)$ to the controller which gives $\dot{\mathbf{q}}(t+1)$. Increment t by 1 and go to step 1.

The online incremental training process does not explicitly have separate training and testing phases. Whenever $\mathbf{y}$ is not given, the robot performs.

Table 2: The results of the leave-one-out test.

	Range	Mean error (with attention)	Mean error (without attention)
θ	$[0, \pi]$	0.090	0.11
v	$[0, 1.0]$	0.005	0.007

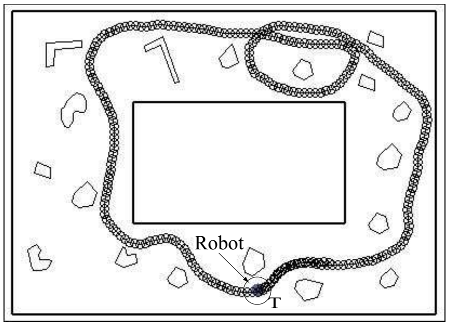

Figure 8: A 10-minute run by the simulated robot with the attentional module. The solid dark lines denote walls and the small trace circles show the trajectory. Obstacles of irregular shapes are scattered about the corridor.

Experimental results

Simulation experiments

To show the importance of the attentional mechanism, two IHDR trees were trained simultaneously: one used attention and the other used the raw range image directly. We interactively trained the simulated robot in 16 scenarios which acquired 1917 samples.

In order to test the generalization capability of the learning system, we performed the leave-one-out test for both IHDR trees. The 1917 training samples were divided into 10 bins. Chose 9 bins for training and left one bin for testing. This procedure was repeated ten times, one for each choice of test bin. The average results are shown in Table 2. Comparing the results, we can see that the mean error was reduced by introducing attention. Although, the generalization capability was improved, the amount is not very large because the training sample consisted of mostly far-away objects.

The tests were performed in an environment different from the training scenarios. In Fig. 8, with attention, the simulated robot performed successfully a continuous 10-minute run. The robot's trajectory is shown by small trailing circles. Remember that no environmental map was stored across the laser maps and the robot had no global position sensors. Fig. 9 shows that, without attention, the robot failed several times in a 3-minute test run.

Experiment on the Dav robot

A continuous 15-minute run was performed by Dav in the corridor of the Engineering Building at Michigan State University. The corridor was crowded with high school students, as shown in Fig. 10. Dav successfully navigated in this dynamic changing environment without collisions with moving students. It is worth noting the testing scenarios were not the same as the training scenarios.

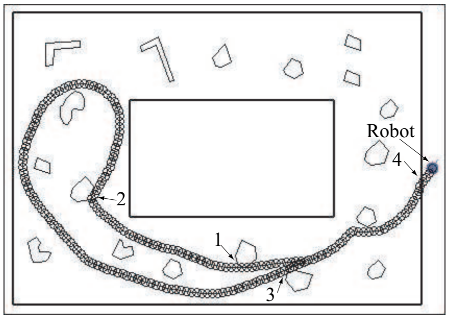

Figure 9: The result of the test without attentional selection. Four collisions indicated by arrows are occurred.

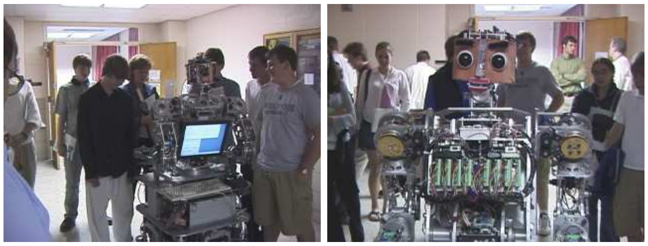

Figure 10: Dav moved autonomously in a corridor crowded with people.

Discussion and Conclusion

The system may fail when obstacles were outside the field-of-view of the laser scanner. Since the laser scanner has to be installed at the front, nearby objects on the side are not "visible." This means that the trainer needs to "look ahead" when providing desired control signals so that the objects are not too close to the "blind spots." In addition, the attentional mechanism assumes that far-away objects were not related to the desired control signal. This does not work well for long term planning, e.g., the robot may be trapped into a U-shape setting. This problem can be solved by integrating this local collision avoidance with a path planner, but the latter is beyond the scope of this paper.

This paper described a range-based obstacle-avoidance learning system implemented on a mobile humanoid robot. The attention selection mechanism reduces the importance of far-away objects when nearby objects are present. The power of the learning-based method is to enable the robot to learn very complex function between the input range map and the desired behavior, such a function is typically so complex that it is not possible to write a program to simulate it accurately. Indeed, the complex range-perception based human action learned by $\mathbf{y} = f(\mathbf{z}, \mathbf{v})$ is too complex to write a program without learning. The success of the learning for high dimensional input $(\mathbf{z}, \mathbf{v})$ is mainly due to the power of IHDR, and the real-time speed is due to the logarithmic time complexity of IHDR. The optimal subspace-based Bayesian generalization enables quasi-optimal interpolation of behaviors from matched learned samples. The online incremental learning is useful for the trainer to dynamically select scenarios according to the robots weakness (i.e., problem areas) in performance. It is true that training needs extra effort, but it enables the behaviors to change according to a wide variety of changes in the range map.

Appendix A

Proof of Theorem 1: Consider the cases where there are k ($1 \leq k \leq h$) end points located within the half-circle T (see Fig. 4). The number of possible configurations for the $h-k$ end points outside the circle, denoted by s_k, is:

$$s_k = q^{h-k}. \tag{11}$$

Because the radial distance of the $h-k$ end points have the freedom to choose values from the interval $[T, r_m]$, which has q discrete values. By definition:

$$E_{P'}\{\mathcal{R}(P')|C\} = \sum_{k=1}^{h} s_k P(k|C),$$

where $P(k|C)$ denotes the conditional probability when k end points are located within the half-circle T. We can see

$$P(k|C) = C_h^k (1-p)^k / (1-p^h).$$

Therefore,

$$\begin{aligned} E_{P'}\{\mathcal{R}(P')\} &= \sum_{k=1}^{h} q^{h-k} \frac{C_h^k (1-p)^k}{1-p^h} \\ &= \frac{\sum_{k=0}^{h} C_h^k q^{h-k} (1-p)^k - q^h}{1-p^h} \\ &= \frac{(q+(1-p))^h - q^h}{1-p^h} > \frac{(1-p)hq^{h-1}}{1-p^h}. \end{aligned}$$

In the last step, the inequality, $(x+\delta)^n - x^n > nx^{n-1}\delta$ if $0 < \delta << x$, is used.

References

Chen, S., and Weng, J. 2000. State-based SHOSLIF for indoor visual navigation. *IEEE Trans. Neural Networks* 11(6):1300–1314.

Fox, D.; Burgard, W.; and Thrun, S. 1997. The dynamic window approach to collision avoidance. *IEEE Robotics and Automation Magazine* 4(1):23–33.

Han, J.; Zeng, S.; Tham, K.; Badgero, M.; and Weng, J. 2002. Dav: A humanoid robot platform for autonomous mental development. In *Proc. IEEE 2nd International Conference on Development and Learning (ICDL 2002)*, 73–81.

Hwang, W., and Weng, J. 1997. Vision-guided robot manipulator control as learning and recall using SHOSLIF. In *Proc. IEEE Int'l Conf. on Robotics and Automation*, 2862–2867.

Hwang, W. S., and Weng, J. 2000. Hierarchical discriminant regression. *IEEE Trans. Pattern Analysis and Machine Intelligence* 22(11):1277–1293.

Pomerleau, D. A. 1991. Efficient training of artificial neural networks for autonomous navigation. *Neural Computation* 3(1):88–97.

Simmons, R. 1996. The curvature-velocity method for local obstacle avoidance. In *IEEE International Conference on Robotics and Automation ICRA'96*, 2275–2282.

Zeng, S.; Cherba, D. M.; and Weng, J. 2003. Dav developmental humanoid: The control architecture and body. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 974–980.