Learning to Press Doorbell Buttons

Vladimir Sukhoy, Jivko Sinapov, Liping Wu, and Alexander Stoytchev
Developmental Robotics Laboratory
Iowa State University
{*sukhoy, jsinapov, lipingwu, alexs*}*@iastate.edu*

***Abstract*— This paper describes an approach that a robot can use to learn to press doorbell buttons. This approach combines exploratory behaviors with an active learning strategy to enable the robot to learn faster how and where it should press a button in order to trigger the buzzer. The framework was tested with an upper-torso humanoid robot on seven different doorbell buttons. Three different active learning exploration strategies were evaluated: random, stimulus-driven, and uncertainty-driven. The results show that an active learning strategy can significantly speedup the robot's learning progress. Among the three strategies that were evaluated, the uncertainty-driven strategy was the most effective.**

I. Introduction

Buttons are ubiquitous in human-inhabited environments. These simple 1-dof widgets are used to control many essential devices and mechanisms. A robot that cannot press buttons would not be able to use anything controlled by a button and thus would not be very useful.

Pressing buttons is challenging for robots for two reasons: 1) human and robot fingers differ; and 2) human and robot senses differ. These difficulties are complicated by the fact that buttons are designed exclusively for humans. Both their appearance and their structure are optimized to make human use more pleasant and efficient. A button that perfectly fits most human fingers may be too small for a robot. Even medium-sized buttons can be difficult to press as robotic fingers made of brushed aluminium slide easily over concave plastic buttons. Furthermore, the feedback that a button generates is intended primarily for humans: keyboard keys click, elevator buttons light up, doorbell buttons ring. A robot without a microphone cannot confirm that it has pressed a doorbell button successfully. It is not realistic to expect that one can write a program that can handle all of these issues without some adaptation mechanism that takes into account the embodiment limitations of the robot.

This paper describes a framework that a robot can use to learn to press buttons autonomously. A large-scale experimental study with an upper-torso humanoid robot (see Fig. 1) was conducted to evaluate the framework. The robot explored seven doorbell buttons and perceived the proprioceptive, auditory, and visual feedback during these interactions. Even though the experiments were performed with doorbell buttons, which provide auditory feedback, the framework can be used with other buttons as well.

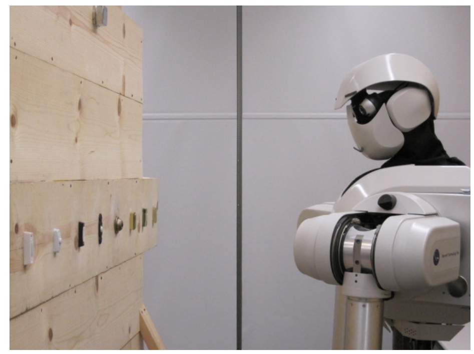


Fig. 1. The upper-torso humanoid robot used in the experiments, shown here looking at the experimental fixture with the buttons.

This work evaluates three different exploration strategies that a robot can use to learn to press buttons faster: random, stimulus-driven, and uncertainty-driven. Under the random strategy the robot did not use what it learned in the past to change the behaviors that it performed. Under the stimulus-driven exploration strategy the robot performed a pushing behavior that was most likely to trigger the buzzer based on prior experience. Under the uncertainty-driven exploration strategy the robot performed a pushing behavior for which it was most uncertain whether it will trigger the buzzer or not.

The robot tested the performance of each of the three strategies on two learning tasks: 1) predicting if a behavior will trigger the buzzer; and 2) predicting whether pushing a button in a specific location in visual space will trigger it. For these two tasks it was estimated how much training is required for the robot to achieve a given performance level. The results indicate that for both learning tasks the uncertainty-driven exploration strategy is the most efficient of the three strategies. Using this strategy, the robot achieved its best performance after 50-100 trials. This fact shows that the learning framework works in real time.

II. Related Work

A. Developmental Psychology

Hauf and Ascherleben [1] demonstrated that 9 months old human infants anticipate the acoustic and visual events associated with pressing different colored buttons. The broader goal of their work was to show that infants use anticipation of action outcomes to control their actions. Hauf et al. [2] found that human infants become more interested in an object after they have had the chance to play with it. They observed that 7-11 month old infants are indifferent to objects or people who manipulate these objects before they have played with the objects. After playing, observing how people manipulate the objects became interesting for the infants. E.J. Gibson [3] showed that humans use observations obtained from active exploration as one of the key sources of knowledge about the world and, in particular, about the affordances of objects.

While our approach is inspired by research studies in Developmental Psychology, learning outcomes are not comparable between the robot and an infant. The robot's behavioral repertoire and the model for behavior outcomes were selected specifically to learn to press buttons. Infants have much more general goals, much more sophisticated behaviors, and much more general models for interpreting their outcomes.

B. Robotics

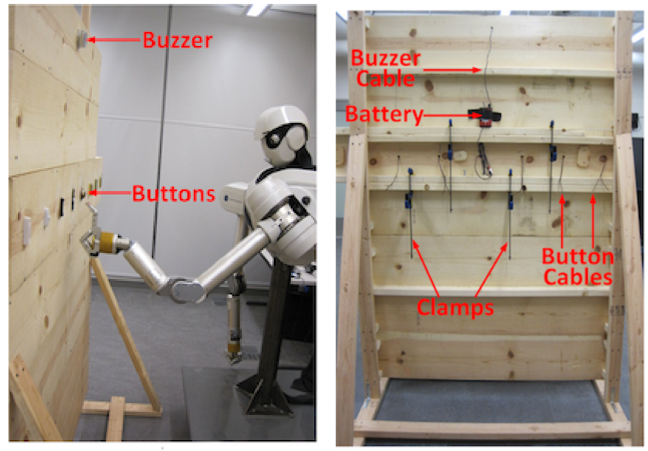
Previous research on pressing buttons in robotics can be divided into three principal categories based on the specific problems they focus on.

1) *Detecting buttons is hard, but pressing them is easy.* Work in this category presupposes that the physical act of pressing buttons is straightforward once they are detected. A key characteristic of approaches from this category is that the button feedback – either tactile sensations from the click itself or feedback from the device controlled by the button – is ignored. The main focus is on training computer vision models for detecting buttons before the robot has attempted to press even a single button. Typically, the papers in this category report results for detecting elevator buttons [4] [5] [6]. In some cases, the detection performance could be improved using the assumption that elevator buttons are arranged in a grid pattern [5] [6].

2) *Both pressing and detecting buttons is hard.* Approaches from this category do not attempt to press or detect buttons as they are. Instead, these approaches seek to modify the environment to simplify the task of programming the robot. For example, reflective markers [7] or RFID tags [8] can be attached to buttons. These tags contain meta-information that informs the robot how and where to press the button and what would happen if it is pressed. The main focus of this research is on possible robotic applications in the home and environmental augmentations that enable them [8].

3) *Button pressing as a social learning task.* Research in this category has focused on pressing buttons in the presence of humans. It was shown that a robot can interpret human-provided cues in order to press a button [9]. It was also shown that a robot can learn to press a button from human demonstrations [10]. The main focus of these studies was on learning to detect and interpret human social cues and not on learning to press buttons autonomously.

In contrast with previous approaches, our work focuses on the physical task of pressing a button. The problem is solved without a prior visual model for detecting a button, without relying on human-provided cues, and without environmental augmentations. Instead, the robot learns to press a button autonomously by analyzing feedback generated by the button as the robot performs exploratory behaviors.



(a) The robot pushing a button. (b) Experimental fixture (back).

Fig. 2. The experimental setup.

Our robot used active learning exploration strategies to learn to press the buttons. Similar strategies have been formulated in previous work using intrinsically-motivated reinforcement learning [11] [12] [13], and POMDPs [14]. Active learning has been used successfully for grasping objects [15] [16] [17]. In contrast with previous applications, the exploration strategies in our work are applied to a robotic manipulation task in the real world and operate on the information extracted from multiple modalities in real time.

III. Experimental Setup

A. Robot

The experiments were performed with the upper-torso humanoid robot shown in Fig. 2. The robot's arms were two Barrett Whole Arm Manipulators (WAMs). The end effector of each arm was a BH8-262 Barrett Hand. A color marker was attached to the tip of the robot's finger to simplify its visual tracking (see Fig. 3).

B. Buttons and Fixture

The robot experimented with 7 doorbell buttons, which were mounted on a wooden fixture (see Fig. 2). The middle segment of the fixture could slide horizontally so that a different button could be presented to the robot during various trials. Behind the wall, the currently explored button was connected to a buzzer and a battery that powered it (see Fig. 2(b)). The buttons were selected from the ones available in stock at a local Lowe's store (a home improvement store).

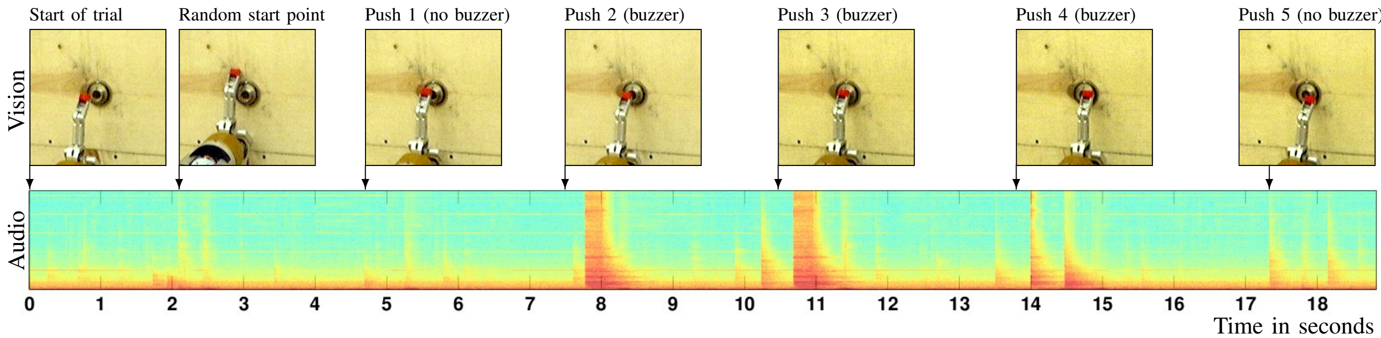


Fig. 3. One of the trials performed by the robot. The spectrogram of the sound is matched to the corresponding video frames for each of the five pushing behaviors. The robot's field of view is larger than the images shown here, which were cropped to show only the area around the button.

C. *Experimental Trials*

The robot used 3 different exploration strategies, which are described in Section IV-A. For each strategy the robot performed 200 experimental trials with each button. In addition to that, the robot collected a test set of 400 trials. The test set was collected using the random strategy described in Section IV-A and the data was used exclusively for evaluating the performance of the 3 strategies. Thus, the robot performed $3 \times 200 + 400 = 1000$ experimental trials with each of the 7 buttons, or 7000 total trials.

Each trial consisted of 5 pushing behaviors directed at the button or the area around it (see Fig. 3). The starting position for each push was the end position of the previous push. The end point of the push was selected based on the learning strategy that was used. To randomize the starting position of each trial, the robot started with a random push that was not counted toward the 5 pushes in the trial. Each trial lasted for approximately 18-20 seconds (the setup time was $3-5$ seconds and each behavior took $\sim$3 seconds). All 1000 trials with each button were performed one after another without interruption. The data collection for each button took approximately 6 hours.

Another way to describe this dataset is as follows: the robot performed 5 pushes $\times$ 1000 times $\times$ 7 buttons $=$ 35000 pushes. This large dataset was necessary for the proper evaluation of the three learning strategies. As described below, the robot learned to press even the most challenging buttons in far fewer trials.

D. *Sensory Data*

During the experiments the robot recorded visual, auditory, and proprioceptive data. The data was processed in real time, but it was also stored to disk for additional offline analysis. Vision data was recorded at 10 frames per second from the robot's left eye (a Logitech QuickCam Pro 4000 webcam) at 640×480 resolution. An Audio-Technica U853AW Hanging Microphone, mounted in the robot's head, was used to record audio at 44.1 KHz. Proprioceptive data, in the form of joint position and torque readings, was recorded from the left WAM arm at 500 Hz.

As the robot performed the experiments, it detected and timestamped "interesting" proprioceptive and auditory events. If the torque magnitude exceeded a predefined limit, the robot recorded a proprioceptive event, interrupted the current pushing behavior, and started the next one. Besides providing behavioral boundaries, proprioceptive events were also used for predicting the outcomes of pushing behaviors in visual space, as described in Section V-D.2.

The robot recorded an auditory event when it heard the buzzer. This was done by analyzing the spectrograms of candidate regions in the audio stream, which were selected if the volume of the audio exceeded a predefined threshold. The maximum duration of a region was set to 0.1 seconds. If the robot detected more than one candidate region for any behavior, it recorded an auditory event only for the first of these regions. To limit the computational load, candidate regions shorter than 0.03 seconds were ignored. The Discrete Fourier Transform (DFT) was calculated for every region to obtain a spectrogram. A frequency component histogram with 20 bins was calculated for every spectrogram and used as an input to a cascade of two classifiers, which decided if a region contained a buzzing sound. The cascade consisted of the Naive Bayes and the K^* classifier [18]. The structure of the cascade was determined from a separate pilot study, in which a training set with "buzzer" and "non-buzzer" sounds was labeled manually.

E. *Behavioral Parametrization*

The robot started each trial from a fixed reference arm position, for which its fingertip was above a button. To randomize the start of the exploration, the robot randomly pushed in the area around the button at the start of each trial. This push did not count towards the exploration, even if it happened to trigger the buzzer. Next, the robot performed 5 exploratory pushing behaviors in the vicinity of the button. The robot started each of these behaviors from the end position of the previous push, and selected the endpoint using the current exploration strategy as described in Section IV-A. To finish the current trial and prepare for the next one, the robot moved its finger back to the reference arm position.

Each pushing behavior consisted of two arm movements: a backup and a following push. The purpose of the backup movement was to enable the robot to push a button even if it missed it during previous pushes. Some of the buttons protruded from the board, so the robot could not push them without a backup movement once it missed them. The backup was always directed at the fixed reference arm position.

Each behavior was parametrized by a vector $x \in \mathbb{R}^6$, which was a concatenation of two vectors $x^{(b)}$ and $x^{(p)}$ in the robot's Cartesian space. $x^{(b)}$ was the direction of the backup movement after the finger hit the button, or the wall, and $x^{(p)}$ was the direction of the subsequent push. Even though the reference arm position was fixed, the parametrization was different for behaviors that differed by their start or end positions on the surface of the experimental fixture.

To represent the result of each behavior the robot used a set of outcomes $O = \{buzzer, no\ buzzer\}$. The robot recorded a $buzzer$ outcome when an auditory event was observed during a behavior. Otherwise, a $no\ buzzer$ outcome was recorded.

IV. Learning Methodology

Over the course of interacting with a button, the robot recorded labeled data points (b_i, o_i), which indicate that outcome o_i was observed when performing a behavior parametrized by the vector b_i. Using this data, a predictive model M was incrementally trained to estimate the conditional probability $P(o|b)$ of observing outcome o given behavioral parameters b.

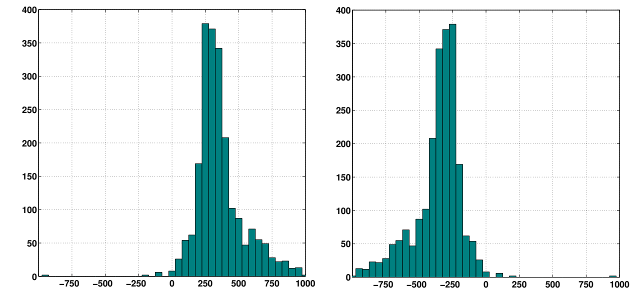
A. Exploration Strategies

Because the predictive model M is updated incrementally, the robot's controller can use it to choose the parameters of the next pushing behavior. To do that, after each behavior, a candidate set $x_1, \ldots, x_N \in \mathbb{R}^6$ is generated, which represents a set of possible behaviors that the robot can perform. The value of N was set to 25. Three different exploration strategies that select the next behavior $x_j \in \{x_1, \ldots, x_N\}$ were evaluated.

1) Random Exploration: Under this exploration strategy, the robot always picks the first candidate in the set $\{x_1, \ldots, x_N\}$, i.e., $j = 1$. Because the candidate set is generated randomly, this strategy results in a random exploration.

2) Stimulus-Driven Exploration: The second exploration strategy selects the behavior from the candidate set that maximizes the expected likelihood of detecting a buzzer sound, as estimated by the robot's predictive model M. In other words, a candidate behavior that is likely to push the button and cause the buzzer to go off is favored over one that is not. Formally, this strategy selects the candidate behavior $x_j \in \{x_1, \ldots, x_N\}$ such that:

$$j = \underset{i=1,\ldots,N}{\operatorname{argmax}} \Pr(buzzer|x_i).$$



(a) Auditory-proprioceptive (b) Proprioceptive-auditory

Fig. 4. Histograms of the temporal intervals between auditory events and proprioceptive events (a) and vice versa (b). These were calculated over all 5000 pushing behaviors that the robot performed with button 1. The bin size of each histogram is 50 ms. The histograms show that the robot first hears the buzzer and then feels that the button is fully pressed.

3) Uncertainty-Driven Exploration: The last exploration strategy selects the behavior, for which the robot's model M is most uncertain regarding its outcome. For each candidate behavior x_i, the uncertainty is quantified using the entropy of the conditional distribution over the set of outcomes in O. Formally, the uncertainty-driven strategy selects behavior x_j according to:

$$j = \underset{i=1,\ldots,N}{\operatorname{argmax}} \sum_{o \in O} -\Pr(o|x_i) \log(\Pr(o|x_i)).$$

B. Predictive Model

A k-NN classifier with $k = 5$ was used to implement the predictive model M that estimated the conditional probability distribution over outcomes given the behavioral parameters. Given $x \in \mathbb{R}^6$ as input, the classifier estimated the conditional probability $\Pr(o|x)$ from the distribution of outcomes of the k nearest neighbors to x in the robot's prior experience, which consists of (b_i, o_i) tuples. For example, if 3 out of 5 nearest neighbors of x have the $buzzer$ outcome, then the probability estimate for $\Pr(buzzer|x)$ is $3/5$.

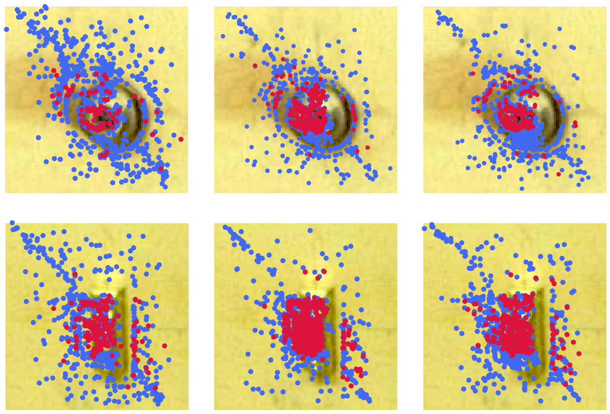
Two key factors motivated the choice of this classifier. First, it can be updated incrementally as more training instances become available. Second, the k-NN classifier was among the best-performing classifiers for a similar task during a pilot study.

V. Results

A. Temporal Intervals between Modalities

For the trials in which the buzzer went off, the robot measured the temporal intervals between proprioceptive and auditory events. Fig. 4 shows a histogram of these temporal delays for one of the buttons. Fig. 4(b) is a mirror copy of Fig. 4(a), which means that an auditory event is almost always followed by a proprioceptive event. The opposite is not true (see Fig. 5).

In general, the robot first heard the buzzer and then detected that it could not press the button any further. The



(a) Random (b) Stimulus-Driven (c) Uncertainty-Driven

Fig. 5. Auditory (red) and proprioceptive (blue) events localized in space for each of the three learning strategies for button 3 (top row) and button 7 (bottom row).

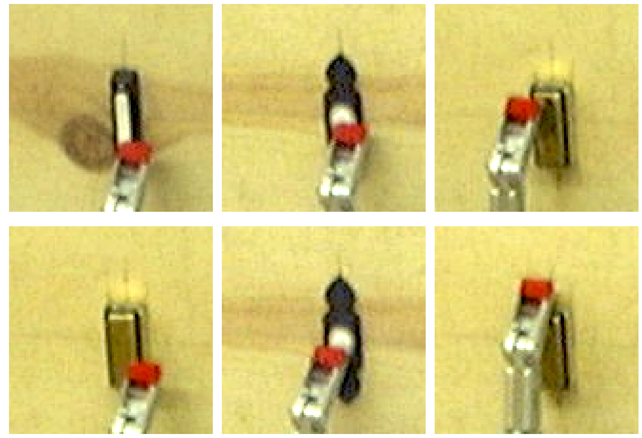


Fig. 6. Video frames from the dataset that show different ways in which the robot was able to ring the doorbell buttons. Note that the red marker on the fingertip does not have to overlap with the pressable part of the button.

average temporal delay was consistently around 250ms. This was true for all buttons as shown in Fig. 8(d). Some buttons were harder to press, e.g., buttons 3 and 4, for which the surface area of the functional part of the button was smaller than that for the other buttons. For these buttons there were fewer entries in the bins of their histograms.

B. Localizing Events in Space

The robot combined the timestamps of the video frames with the timestamps associated with auditory and proprioceptive events. In this way, the robot's fingertip could be localized in visual space during these events. The spatial distributions of these events for the best performing exploration strategy for all buttons are shown in Fig. 8(b) and 8(c). To highlight the differences in the spatial distributions of events for different learning strategies, these distributions are shown for two of the buttons for all three strategies in Fig. 5. Both active learning strategies produced more auditory events than the random strategy for the same number of pushing behaviors.

In many cases the robot pressed buttons with parts of its finger other than its fingertip (see Fig. 6). This and the fact that visual tracking was done only for the fingertip explain the apparent scattering of the points in Fig. 5, 8(b), and 8(c). Finally, the parameters for the pushing behaviors were sampled uniformly in joint space, which resulted in less than uniform sampling in Cartesian and visual space. In particular, the main diagonal was oversampled, as can be seen in Fig. 5. Diagonal oversampling did not influence the auditory event locations significantly, which can be seen in Fig. 8(c).

C. Performance Measures

The performance was measured on the task of predicting whether a pushing behavior would trigger an auditory event. The k-NN classifier with $k = 5$ generated these predictions given either the behavioral parameter vector $x \in \mathbb{R}^6$ or the 2D tracking coordinates of the robot's fingertip. The predictions were compared against the "ground truth" – whether the robot detected the buzzer sound during this behavior or not.

The robot performed 400 trials to collect the evaluation set. Each trial consisted of 5 random pushing behaviors. Thus, there were 2000 pushing behaviors in the test set for each button. Out of the 2000 test behaviors for button 3, the buzzer sound was detected during only 70 of them. A naive approach can predict that no buzzer was detected for all 2000 behaviors and reach a 96.5% accuracy. These predictions, however, would be useless despite the high accuracy.

A performance measure that is more adequate in this case is the Cohen's kappa coefficient [19], which compares the model accuracy against chance accuracy as follows:

$$\kappa = \frac{\Pr(a) - \Pr(e)}{1 - \Pr(e)},$$

where $\Pr(a)$ is the raw accuracy, i.e., the probability of a correct prediction by the classifier, and $\Pr(e)$ is the probability of a correct prediction by chance. In our example, $\Pr(a) = \Pr(e) = \frac{1930}{2000}$, and $\kappa = 0$. Another example: suppose that there were 52 true positives, 30 false positives, 18 false negatives, and 1900 true negatives. In this case, $\Pr(a) = \frac{52+1900}{2000} = 0.976$, $\Pr(e) = \frac{70}{2000} \cdot \frac{82}{2000} + \frac{1930}{2000} \cdot \frac{1918}{2000} = 0.92686$, and $\kappa \approx 0.67$. Thus, the kappa coefficient measures the performance more adequately than the raw accuracy.

There are no universally accepted rules for interpreting the numerical values of the κ coefficient. Landis and Koch [20] suggested metrics based on personal opinion (see Table I). According to these metrics, the performance for predictions in proprioceptive space shown in Fig. 8(e) mostly falls into the "moderate" category, while the performance for predictions in visual space (see Fig. 8(f)) mostly falls into the "substantial" category.

κ	Strength of Agreement
0.81 – 1.00	Almost Perfect
0.61 – 0.80	Substantial
0.41 – 0.60	Moderate
0.21 – 0.40	Fair
0.00 – 0.20	Slight
< 0	Poor

TABLE I

TABLE FOR INTERPRETING κ COEFFICIENT VALUES PROPOSED BY [20].

D. Learning to Predict Behavioral Outcomes

The robot used the collected data to predict if a pushing behavior would trigger an auditory event. For each of the three learning strategies, and for each of the buttons, the number of available learning trials was varied to find out how much training is required to achieve a specific performance level. The predictions were evaluated on the 400 trials from the testing set.

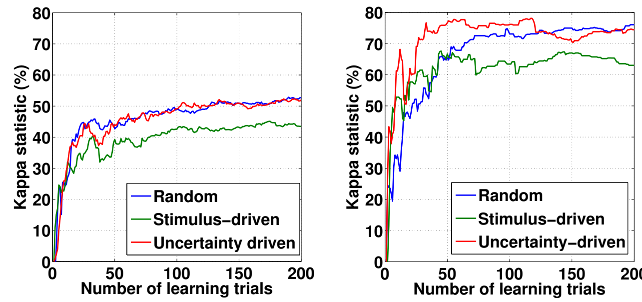
1) Using Proprioceptive Behavioral Parameters: The robot trained the same model based on the k-NN classifier that was used by the learning strategies (see Section IV-A) to estimate the conditional probability distribution of outcomes given the behavioral parameter vector $x \in \mathbb{R}^6$. For button 2, the results of this evaluation are shown in Fig. 7(a). The results for all buttons are shown in Fig. 8(e). For all buttons the model trained for the uncertainty-driven exploration strategy was either the top performer or one of the top performers.

For button 3 – the hardest button to press – all strategies failed to achieve a decent learning performance in 200 training trials. This suggests that predicting behavioral outcomes using only behavioral parameters in proprioceptive space is not always possible. Using other modalities and more sophisticated learning approaches may improve learning performance, as shown below.

2) Using Visual Press Locations: Fig. 4 shows that auditory events are shortly followed by proprioceptive events. Therefore, the question arises: How much training does the robot need to learn to tell if the buzzer will sound if the button is pressed at a specific location in visual space?

To answer this question, a model based on the k-NN classifier with $k = 5$, was trained to estimate the conditional probability distribution of behavioral outcomes given the 2D position of the robot's finger in visual space when the proprioceptive event occurred. Behaviors that produced no proprioceptive events were not considered for learning or evaluation. This model was similar to the predictive model used in Section IV-A and Section V-D.1.

The learning performance in visual space is substantially better than in proprioceptive space. The learning problem in proprioceptive space, however, is substantially more difficult as the location of the push in visual space is unknown at the start of the behavior and depends on both the button and the behavior. This difference in problem complexity explains the difference in learning performance.



(a) Proprioceptive Space (b) Visual Space

Fig. 7. Learning curves for predicting if an auditory event will be heard given: (a) the kinematic parameters of a pushing behavior in proprioceptive space; and (b) the location of the robot's fingertip in visual space. The results are for button 2.

The evaluation results are shown for one button in Fig. 7(b) and for all buttons in Fig. 8(f). For all buttons the uncertainty-driven active learning strategy performed the best. For some of the buttons it was the best by a wide margin. With the uncertainty-driven active learning strategy approximately 50-100 trials were sufficient to achieve the best performance for most of the buttons.

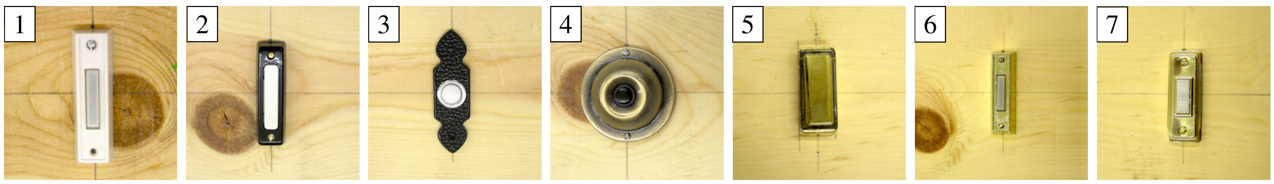
VI. CONCLUSIONS AND FUTURE WORK

This paper described a framework that a robot can use to learn to press buttons. The key input for learning this task was the multimodal feedback perceived by the robot as it performed exploratory behaviors directed at the buttons. The framework was tested on doorbell buttons, which generated auditory feedback, but it can be applied to other types of buttons as well.

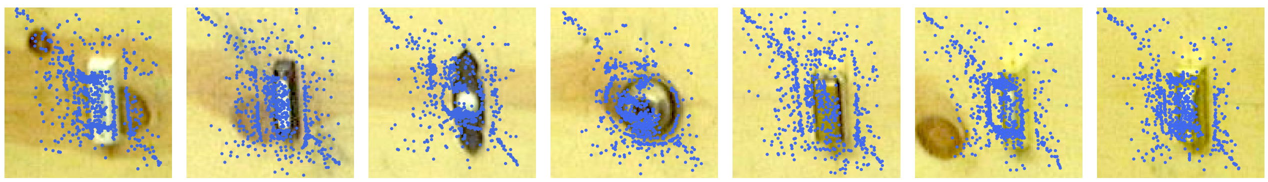
This paper evaluated three strategies that a robot can use to learn how to press buttons in real time: random, stimulus-driven, and uncertainty-driven. The random strategy always selected a random push. The stimulus-driven strategy selected a behavior that was most likely to press the button. The uncertainty-driven strategy selected the behavior with the most uncertain outcome.

Each of the three learning strategies was evaluated on two tasks: 1) predicting if pressing the button at a specific spatial location would cause the buzzer to go off; and 2) predicting if a behavior would successfully press a button. Overall, the uncertainty-driven exploration strategy learned faster than each of the alternatives for both tasks. For the "hardest" button in the dataset (button 3) both active learning strategies learned much faster than the random strategy in visual space. Similar results were observed in proprioceptive space for button 4 (the one with the smallest pressable part). For both learning tasks, and for most buttons, the uncertainty-driven strategy achieved its best performance in 50-100 trials.

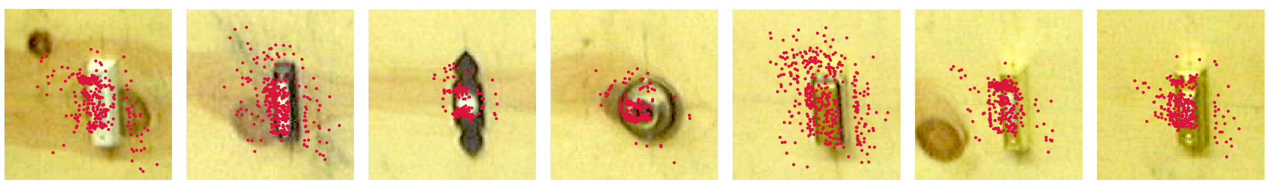
In our opinion, there are two reasons for the superior performance of the uncertainty-driven strategy: 1) the other two



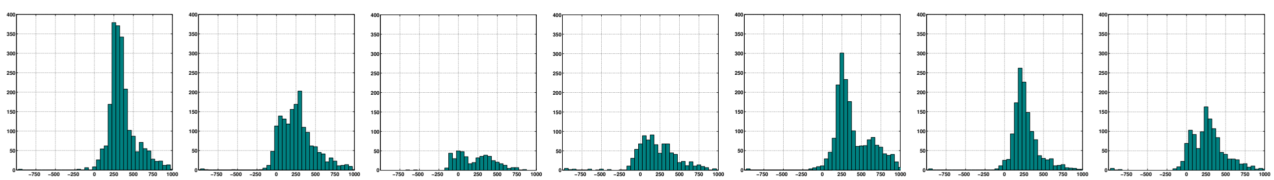
(a) The seven doorbell buttons explored by the robot.



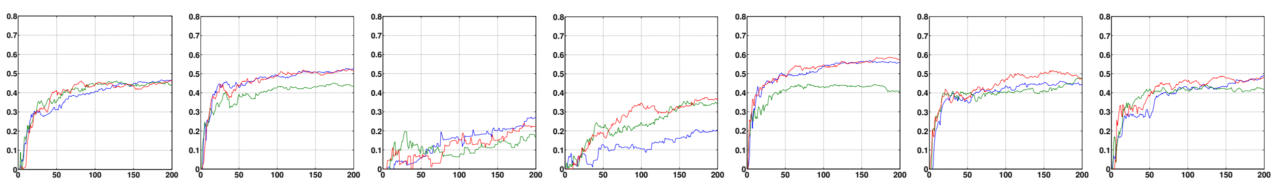
(b) Proprioceptive events localized in visual space for the uncertainty-driven exploration strategy over 200 trials (1000 pushes).



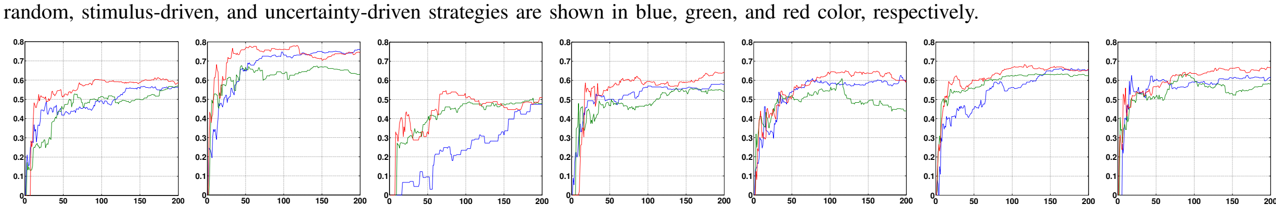
(c) Auditory events localized in visual space for each of the buttons using the uncertainty-driven strategy.



(d) Histograms of the temporal intervals between auditory and proprioceptive events for each button. The bin size of each histogram is 50 ms.



(e) Learning curves for the task of predicting if a given pushing behavior will trigger an auditory event as a function of the number of learning trials (x-axis). The performance measure (y-axis) is the kappa statistic estimated from a separate evaluation set of 400 trials. The curves for the random, stimulus-driven, and uncertainty-driven strategies are shown in blue, green, and red color, respectively.



(f) Learning curves for the task of predicting if any behavior for which a proprioceptive event is detected in a specific region of visual space will trigger an auditory event as a function of the number of learning trials (x-axis). The performance measure (y-axis) is the kappa statistic, which was estimated from a separate evaluation set of 400 trials. The curves for the random, stimulus-driven, and uncertainty-driven strategy are shown in blue, green, and red color, respectively.

Fig. 8. Summary of the experimental results for all buttons. See the text for more details.

strategies do not take the overall structure of the predictive model into account; and 2) picking an action with the most uncertain outcome is an approximation for minimizing the uncertainty of future queries to the predictive model. Indeed, the random strategy does not really explore as its choices do not depend on past observations. The stimulus-driven strategy focuses only on those behaviors that trigger the buzzer – it is not interested in the overall structure of the predictive model. Only the uncertainty-driven strategy uses the overall structure of the predictive model for both positive and negative outcomes. Future work may compare the uncertainty-driven strategy with more sophisticated strategies that are specifically focused on minimizing the uncertainty of future queries to the predictive model, e.g., a strategy that plans future behaviors to achieve this.

The framework already can localize auditory and proprioceptive events in visual space as the robot explores a button. Future work can establish a correlation between the visual features of the button and the spatial distributions of sensory events. The robot can use this information to learn a visual model that can segment an image of a button into functional parts. For example, the robot can use the extracted visual features to identify the functional parts of novel buttons without the need to explore them first. Modifying the methodology to work with light switches is yet another feasible direction for future work.

ACKNOWLEDGEMENTS

The authors would like to thank Eric Everett, Michael Sambol, and Timothy Kalpin who assembled the experimental fixture with the doorbell buttons.

REFERENCES

[1] P. Hauf and G. Aschersleben, "Action-effect anticipation in infant action control," *Psych. Research*, vol. 72, no. 2, pp. 203–210, 2008.

[2] P. Hauf, G. Aschersleben, and W. Prinz, "Baby do-baby see!: How action production influences action perception in infants," *Cognitive Development*, vol. 22, no. 1, pp. 16–32, 2007.

[3] E. Gibson, "Exploratory behavior in the development of perceiving, acting, and the acquiring of knowledge," *Annual review of psychology*, vol. 39, no. 1, pp. 1–42, 1988.

[4] J. Miura, K. Iwase, and Y. Shirai, "Interactive teaching of a mobile robot," in *Proceedings of ICRA*, 2005, pp. 3378–3383.

[5] K.-T. Song and T.-Z. Wu, "Visual servo control of a mobile manipulator using one-dimensional windows," in *Proceedings of Industrial Electronics Society*, vol. 2, 1999, pp. 686–691.

[6] E. Klingbeil, B. Carpenter, O. Russakovsky, and A. Y. Ng, "Autonomous operation of novel elevators for robot navigation," in *Proceedings of ICRA*, 2010, pp. 751–758.

[7] R. Katsuki, J. Ota, Y. Tamura, T. Mizuta, T. Kito, T. Arai, T. Ueyama, and T. Nishiyama, "Handling of objects with marks by a robot," in *Proceedings of IROS*, vol. 1, 27-31 2003, pp. 130–135.

[8] H. Nguyen, T. Deyle, M. Reynolds, and C. Kemp, "PPS-tags: Physical, perceptual and semantic tags for autonomous mobile manipulation," in *Proceedings of the IROS Workshop on Semantic Perception for Mobile Manipulation*, 2009.

[9] A. Thomaz, "Socially guided machine learning," Ph.D. dissertation, Massachusetts Institute of Technology, 2006.

[10] J. Lieberman, "Teaching a robot manipulation skills through demonstration," Master's thesis, Massachusetts Institute of Technology, 2004.

[11] P.-Y. Oudeyer and F. Kaplan, "What is intrinsic motivation? a typology of computational approaches," *Frontiers in Neurorobotics*, vol. 1, pp. 1–6, 2007.

[12] J. Marshall, D. Blank, and L. Meeding, "An emergent framework for self-motivation in developmental robotics," in *Proceedings of ICDL*, 2004, pp. 104–111.

[13] A. Baranes and P.-Y. Oudeyer, "R-IAC: Robust intrinsically motivated exploration and active learning," vol. 1, no. 3, oct. 2009, pp. 155–169.

[14] N. Butko and J. Movellan, "I-POMDP: An infomax model of eye movement," in *Proceedings of ICDL*, 2008, pp. 139–144.

[15] M. Salganicoff, L. Ungar, and R. Bajcsy, "Active learning for vision-based robot grasping," *Machine Learning*, vol. 23, no. 2–3, pp. 251–278, 1996.

[16] L. Montesano and M. Lopes, "Learning grasping affordances from local visual descriptors," *Proceedings of ICDL*, pp. 1–6, 2009.

[17] A. Morales and E. Chinellato, "Active learning for robot manipulation," in *Proceedings of the European Conference on AI*, 2004, pp. 905–909.

[18] J. Cleary and L. Trigg, "K*: An instance-based learner using an entropic distance measure," in *Proceedings of the 12th International Conference on Machine learning*, 1995, pp. 108–114.

[19] J. Cohen, "A coefficient of agreement for nominal scales," *Educational and Psychological Measurement*, vol. 20, no. 1, pp. 37–46, April 1960.

[20] R. Landis and G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, no. 1, pp. 159–174, 1977.