Modeling robot trust based on emergent emotion in an interactive task

Murat Kirtay
Adaptive Systems Group,
Department of Computer Science,
Humboldt-Universität zu Berlin
Berlin, Germany
murat.kirtay@informatik.hu-berlin.de

Erhan Oztop[1,2], Minoru Asada[1]
[1]*Symbiotic Intelligent Systems Research Center,*
Institute for Open and Transdisciplinary
Research Initiatives, Osaka University, Japan
[2]*Ozyegin University, Istanbul, Turkey*
{erhan.oztop,asada}@otri.osaka-u.ac.jp

Verena V. Hafner
Adaptive Systems Group,
Department of Computer Science,
Humboldt-Universität zu Berlin
Berlin, Germany
hafner@informatik.hu-berlin.de

***Abstract*—Trust is an essential component in human-human and human-robot interactions. The factors that play potent roles in these interactions have been an attractive issue in robotics. However, the studies that aim at developing a computational model of robot trust in interaction partners remain relatively limited. In this study, we extend our emergent emotion model to propose that the robot's trust in the interaction partner (i.e., trustee) can be established by the effect of the interactions on the computational energy budget of the robot (i.e., trustor). To be concrete, we show how high-level emotions (e.g., well-being) of an agent can be modeled by the computational cost of perceptual processing (e.g., visual stimulus processing for visual recalling) in a decision-making framework. To realize this approach, we endow the Pepper humanoid robot with two modules: an auto-associative memory that extracts the required computational energy to perform a visual recalling, and an internal reward mechanism guiding model-free reinforcement learning to yield computational energy cost-aware behaviors. With this setup, the robot interacts with online instructors with different guiding strategies, namely reliable, less reliable, and random. Through interaction with the instructors, the robot associates the cumulative reward values based on the cost of perceptual processing to evaluate the instructors and determine which one should be trusted. Overall the results indicate that the robot can differentiate the guiding strategies of the instructors. Additionally, in the case of free choice, the robot trusts the reliable one that increases the total reward – and therefore reduces the required computational energy (cognitive load)– to perform the next task.**

***Index Terms*—Trust, Emotions, Internal reward, Visual recalling, HRI**

I. INTRODUCTION

Trust is a crucial element to achieve smooth interactions among humans in different settings: playing a team game (e.g., soccer or electronic games), solving a task in a competitive environment (e.g., solving a jigsaw puzzle), to mention a few. Due to its critical role in interactive social learning, the interest in trust for physically and virtually embodied agents attracts researchers in the areas of autonomous and decision support systems, human-robot interaction, and collaboration [1], [2], [3].

Although trust has been an active area of research in robotics, most of the studies (especially in human-robot interaction) focus on determining psychological, environmental, cognitive, and design factors that play an essential role in establishing human trust in artificial agents [4]. However, in this study, we address one of the less-explored aspects of trust in robotics by answering the following question: How could a robot develop trust in an interaction partner (e.g., a human or an online instructor)?

To this end, we designed a sequential visual pattern exploration task for the Pepper robot where it has to recall visual memories based on what it sees with close-to-minimal computational processing cost. The robot is situated in a scenario where it interacts with three types of online instructors, *reliable*, *less reliable*, and *random* by asking suggestion as to what pattern to observe next. For each pattern observed, the robot generates an internal reward signal based on the computational cost of the recalling process that is undertaken by a Hopfield network, which serves as a model for more general cognitive processing [5]. The robot requests help from its instructor to reduce the computational cost when the cost incurred by the current pattern is non-negligible. During an interaction session, the robot applies Reinforcement Learning (RL) and converges to a sequential observation behavior based on its own decisions (RL-exploration) as well as the suggestions given by its instructor. At the end of the experiment, the robot associates cumulative internal rewards it acquires with the instructors as a means to differentiate the trustworthiness of the instructors, which it may exploit when asked to choose an instructor for subsequent interactions – that is, a free choice to select an instructor.

The rest of this paper is organized as follows. In Section II, we review related work on human trust and robot trust. Next, we present an emergent emotion model and its modules –namely an auto-associative memory and internal reward mechanisms– in Section III. In Section IV, we describe the experimental setup and the performed task. Then, we present the results in Section V and provide links to reproduce the results in Section VI. Finally, we provide the conclusions

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2002/1 "Science of Intelligence" - project number 390523135. Additional support is provided by the International Joint Research Promotion Program, Osaka University under the project "Developmentally and biologically realistic modeling of perspective invariant action understanding".

and future directions of the research.

II. RELATED WORK

In this section, we first introduce the works that aim at determining the factors that affect developing human trust in artificial agents. We then present studies that provide models of trust for robots. It is worthwhile to note that the former encompasses almost most of the trust-related studies and only a limited number of studies is available that focus on the latter.

Sanders et al. [6] proposed a framework for the development of human trust in a human-robot team setting. The study introduces environmental, robot, and human factors that play an important role in establishing human trust in robots, including predictive and adaptive behavior of the robot, the mental workload of the human, and social skills of the robot: expressing emotions and turn-taking [6]. Novitzky et al. [7] employed physiological signals (e.g., heart rate variability) to measure the cognitive load of humans while playing a game (i.e., capture the flag). Here the authors hypothesize that reducing the cognitive load will increase trust in the simulated game partner. Ahmad et al. [8] designed a matching pair game to assess the relationship among cognitive load, trust, and anthropomorphism (i.e., the Pepper humanoid robot vs. the Husky mobile robot) in a Human-Robot game setting. The results suggest an inverse proportional relationship between cognitive load, measured by the change of pupil size of the participants and trust. Correia et al. [9] designed a card-playing game in which human card players were partnered with robots and humans. In this study, the authors focused on showing whether the level of trust can be developed differently for humans and robots. Here the human trust in robot partners was measured by filling questionnaires before and after the game. The authors concluded that the trust in the robot is mostly based on performance, whereas, for the human partner, it is related to behavioral response and emotions.

Based on the conclusions drawn from the aforementioned works, we underline that cognitive load and emotions are two factors that substantially affect the development of human trust in agents. Instead of assessing the roles of various factors for humans in artificial systems, we followed a distinct approach compared to the studies mentioned above. In that, we embed high-level emotions (e.g., the well-being of the agent) and cognitive load (i.e., a computational cost of processing a visual stimulus to decide for an action) on the robot platform that develops trust in the online instructor that provides guidance to perform the task.

Next, we introduce the trust studies for simulated and actual robots. Patacchiola and Cangelosi [10] presented a trust model for simulated cognitive agents. In this setting, the simulated agent (equipped with a theory of mind module) interacts with the informants to either help or trick the agent. The results show that the agent with the theory of mind component distinguishes reliable and less reliable informants. As a continuation of this study, the authors employed the same trust module to propose a comprehensive cognitive architecture that enables a humanoid robot to separate different information sources (i.e., reliable vs. less reliable interaction partners) in the context of an object naming experiment and distinguishing helper and tricker informants [11], [12]. Our implementation differs from the work in [10], [11], and [12] in the following ways. On the one hand, we realize our model on the actual robot platform that operates in a non-controlled setting in which environmental dynamics (e.g., noise and hardware calibrations) might affect the results. On the other hand, we did not equip our agent with a higher-level cognitive concept such as theory of mind; instead, we aim to show that robot trust in a reliable instructor could be autonomously established by simple mechanisms: auto-associative memory and internal reward based on computational load.

III. METHODS

We employ two cognitive modules to form robot trust in an interaction partner: auto-associative memory and internal reward for decision making. These modules were developed by applying the formalization provided in [13] and implemented in [14], [5]. We followed the same hypothesis that we introduced in [15]: The emergence of high-level emotions (e.g., well-being) in the neural system of high-level organisms (e.g., humans) or an artificial agent can be formulated by the neurocomputational energy regulation need of the agent.

To formally explain that the robot could establish trust in an instructor, we put forward the following proposal: if interactions with an instructor lead to less amount of energy consumption (or less cognitive load) for a given task (i.e., visual recalling), the agent will develop engaging emotions and therefore it will establish trust in the instructor. Here we refer to cognitive load as the amount of (neuro) computational resources that requires performing a visual recalling task. We note that our proposal is also well aligned with the studies that aim at determining the trust factor of humans in artificial agents. For instance, the studies that we introduce in Section II also emphasize that cognitive load and emotion are two potent elements for humans to establish trust in artificial and biological agents [6], [7], [8], [9].

A. Auto-associative memory

To form an auto-associative memory of the robot, we use a modified version of the Hopfield Network, i.e., High Order Hopfield Network [13]. This module is employed to enable the robot to process a visual stimulus and derive the cost of computation while recalling a visual pattern. We note that depending on the context, the terms *cost of computation*, *computational energy,* and *cognitive load* are used interchangeably throughout the paper.

We train the network by using the images in Figure 1. The robot captures the images through a camera in a sequence, then a preprocessing pipeline (i.e., extracting a region of interest, grayscaling, binarizing, and downsizing the patterns) was performed to obtain the bipolar representation of images with a size of 32×32. To employ the network on the bipolarized (each pixel is ± 1) patterns, the activation (i.e., output representation) of each unit i (i.e., a neuron) is given

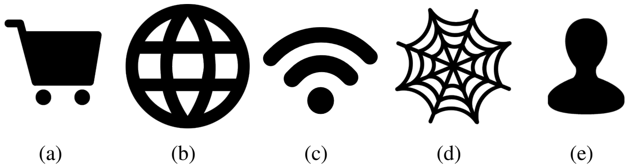


Fig. 1: Visual patterns stored to form an auto-associative memory.

by the sum of the products of the activations of all possible pairs of units:

$$S_i = sgn\left(\sum_{jk} W_{ijk} S_j S_k\right) \tag{1}$$

where $sgn()$ produce -1 for negative arguments and $+1$ for the arguments that are greater than or equal to zero. To derive the weight matrix for the training patterns, we employed Eq. 2:

$$W_{ijk} = \sum_{p} \xi_i^p \xi_j^p \xi_k^p \tag{2}$$

where ξ_i^p, ξ_j^p and ξ_k^p are the ith, jth and kth bipolar bits of the pth pattern ξ^p. Since we used five training patterns as shown in Figure 1, this equation has been performed with $p = 5$.

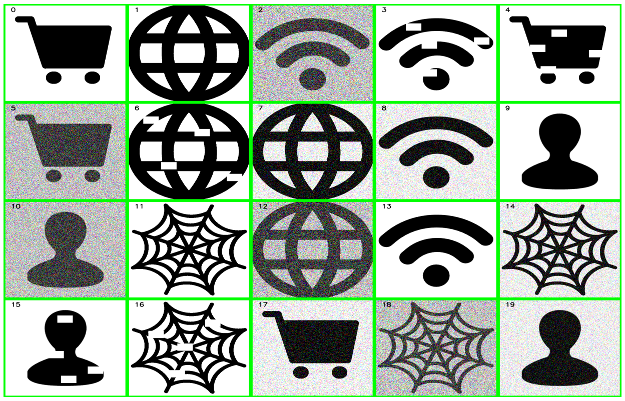


Fig. 2: The constructed visual scene for perceptual processing.

The weight matrix was constructed to perform the visual recalling task. The visual recalling has been performed on the images in the visual scene in Figure 2. Here, the visual scene is composed of 20 patterns in the form of a 4×5 grid where each cell in the grid hosts either one of the trained patterns, the noisy version, or the cropped version of the trained patterns.

In the iteration of the experiment, when the agent selects the id number associated with a specific cell in the grid trough a voice command, the pattern in that cell is displayed in a full-screen mode. Then the robot captures the input image (denoted as ξ) through its camera and performs the preprocessing steps given above, to obtain the bipolar version of the image. Next, the preprocessed image becomes the input or the initial state of the network after which network dynamics is run for visual recalling. The network dynamics is run by asynchronously updating the units to reach a steady-state (i.e., convergence state), shown as $\bar{\xi}$. We noted that the convergence state might be one of the following: a stored pattern, an inverse of a stored pattern, or a combination of stored patterns [16].

We define the neural computational cost of recalling a memory pattern ξ, as the number of changed bits to reach a steady-state pattern, and call it *energy* of ξ denoted by $E(\xi)$. This energy value for a visual pattern is then used to derive an internal reward signal to perform cost-aware decision making.

B. *Internal reward module*

To enable the robot to make cost-aware decisions, we used a model-free reinforcement learning algorithm, namely SARSA. Here we note that the formalization of the internal reward mechanism is adopted from our previous study [15].

The decision-making system was implemented by adopting the Markov Decision Process (MDP) framework [17]. The MDP framework is defined as a tuple (S, A, P, R) that is composed of a set of states (S), a set of actions (A), a state transition function (P), and a reward function (R). In the reinforcement learning setting, the solution of an MDP is an optimal policy (π) that describes what action to take in a given state so that the agent maximizes the sum of rewards it can collect in the long run. Although finite MDPs can be solved efficiently (e.g., by Dynamic Programming), in this study, we assume no model knowledge (i.e., no P) and require on-policy learning ability so that the system can be deployed on robots in more general settings. To be concrete, we adopt the on-policy learning algorithm SARSA with $\varepsilon - greedy$ exploration strategy to update the policy based on the decision of the robot for each iteration.

In our implementation, a state ($s \in S$) was defined as an enumerated discrete region in the scene (as shown in Figure 2 with 20 states, s_i where $i \in (0, n_s - 1)$) that hosts a visual pattern to be displayed to the robot. Then the perceived image is processed to be an input of the auto-associative memory module to extract a cost value (i.e., computational energy) for visual recalling. In this setting, the actions of the agent were grouped into two categories. In the first category, an action ($a \in A$) is performed by a voice command of the agent to perceive a pattern located in the specific state in the scene. In that, the number of actions, n_a, is equal to the number of states, a_i where $i \in (0, n_a - 1)$. To be more specific, the agent can perceive a pattern by generating a voice command associated with the state number, including the current state where it is located.

In the second category, the agent requests the instructor to select an action to make a decision. After the instructor's suggestion, the agent takes action to select a pattern to perform visual recalling. Since the agent acts by its own and employs its auto-associative memory module after interacting with the instructor, we did not modify the MDP framework. To be concrete, the instructor only decides the next state for the agent. Then the agent performs the same procedures for visual recalling and decision making by using the same tuple elements, (S, A, P, R).

The SARSA algorithm performs $\varepsilon - greedy$ strategy to iterate over the state-action pairs. By leveraging this strategy, the agent chooses the action to visit the most valuable state by comparing the values of all available states – i.e., exploitation. Additionally, a fixed rate of randomness (i.e., ε is chosen to be 0.3) was introduced for exploring the environment. The SARSA algorithm performs the update given in Eq. 3 to learn the value of state-action pairs, $Q(s, a)$:

$$Q(s,a) \leftarrow Q(s,a) + \mu(R(s,s') + \gamma Q(s',a') - Q(s,a)) \quad (3)$$

In Eq. 3, $Q(s', a')$ indicates the value for the action a' in the next state s'. The μ variable is the step size (i.e., learning rate) parameter, γ is an adjustment factor that discounts expected future rewards. The μ and γ values are set to 0.7 and 0.4, respectively. We adapt these values to our previous study [15], which were determined by performing a grid-search.

We emphasize that in most reinforcement learning applications, the reward function, R, is often hand-crafted and application-specific. However, in our setting, the agent internally generates a reward value $R(s, s')$ for an s, s' pair. To this end, in Eq. 4, we define the reward as a function of the computational energy required to process a visual pattern.

$$R(s,s') = \begin{cases} -1 & if \ E(\xi^s) < E(\xi^{s'}) \\ \ \ 1 & if \ E(\xi^s) \geq E(\xi^{s'}) \end{cases} \quad (4)$$

The ξ^s and $\xi^{s'}$ are the image patterns received in the states s and s', respectively, and the energy values (i.e., the number of changed bits to reach the stable state) for visual recalling operations are denoted by $E(\xi^s)$ and $E(\xi^{s'})$.

Overall, the internal reward mechanism positively reinforces the agent if it moves the state associated with the higher energy consumption to the lower one. In an inverse situation, the same internal mechanism generates negative reinforcement for the agent. We note that the cumulative reward obtained through this reward function also indirectly includes the effect of the instructor's response pattern towards the agent. That is why we suggest that the cumulative reward is among the metrics that can be used to determine robot trust in the instructor.

IV. EXPERIMENTAL SETUP

The experimental setup consists of the Pepper humanoid robot and monitors for perceptual processing and interacting with online instructors (see Figure 3). The monitor provides a scene divided into a 4×5 grid in which each cell hosts a visual pattern with an integer id ranging from 0 to 19. Here, the robot perceives the state information from the environment – i.e., a monitor that displays a pattern in a full-screen mode for the selected pattern in the grid. To be concrete, the robot selects a specific region in this scene by generating a voice command (e.g., "Open 13"). Then the pattern in that region is presented to the robot as state information for perceptual processing. After preprocessing the perceived visual pattern, the robot employs its auto-associative memory module to extract the cost of perceptual processing– that is, the required computational energy to reach a converged state. The cost values for two consecutive states are compared to derive an internally generated reward value. In that, if the reward value is negative, the agent interacts with the online instructor to decide for the next action. In other words, in a scaffolding setting, the robot requests help from the more knowledgeable other to achieve the assigned task (i.e., visual recalling) with less cognitive load. Otherwise, the agent continues to interact with the environment by taking action by itself. The robot interacts with online instructors for 500 iteration steps starting from a randomly selected initial state. Each iteration is repeated for five runs to account for sensory noise and stochasticity in the implemented modules (e.g., exploration rate). We note that the online instructors were programmed to perform different guiding strategies: reliable, less reliable, and random. The instructor with a random strategy guides the robot to process visual patterns without considering the pattern-energy association. However, the less reliable instructor performs decisions that lead to noisy or partially cropped versions of stored patterns to be processed by the robot. This guiding strategy deliberately aims at increasing the cognitive load required for visual recalling. In contrast, the reliable instructor guides the agent towards one of the memory patterns that the robot has originally stored in its auto-associative network, thus leading to perceptual processing with low energy requirement.

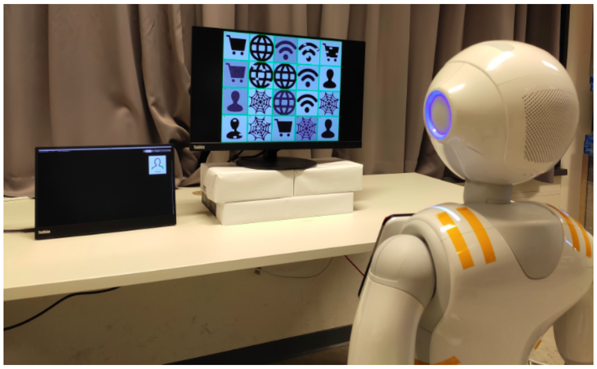
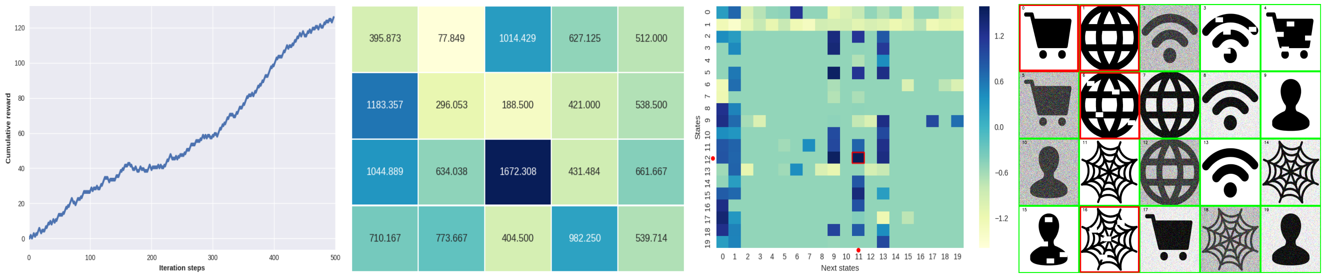


Fig. 3: Experimental setup where the Pepper humanoid robot has to perform visual recalling (i.e., a memory game) on a screen with support of an online instructor.

After performing the visual recalling experiments with each instructor, the robot is provided a free choice to choose an instructor for the next task. In this case, the robot chooses the instructor that decreases the cognitive load of the robot by guiding the robot to increase the average cumulative reward. Here we suggest that this behavior could be associated with developing the robot's trust in a specific instructor.

V. RESULTS

This section provides the results that were obtained by interacting with the three types of instructors: *reliable*, *less-reliable*, and *random*. Before presenting the results for all



(a) Cumulative reward curve (b) Average energy per state (c) Q matrix heatmap (d) Final states

Fig. 4: Experimental results for 500 iterations while interacting with the reliable instructor. The horizontal and vertical axes in Figure 4(a) indicate the cumulative reward and iteration steps, respectively. The vertical axis of Figure 4(c) shows the id numbers of states, and the horizontal axis presents the id numbers of actions (i.e., the next-states). An example of a state-action (or state-next state) pair is illustrated with red dots: state 12 will lead to visiting state 11 as the best available state according to Q value.

iterations, we first visualize the results for one of the runs with the reliable instructor. Although the visualization of the results for the other runs will be different, the same description and visualization procedure will also be valid for all runs with different instructors. In this way, we aim to show how the related metrics were derived and visualized to interpret the results for all runs.

Figure 4 shows the experimental results for the robot while interacting with the reliable instructor for 500 iterations. In this figure, the cumulative reward curve of the agent is illustrated in Figure 4(a). The increasing trend in this curve underlines that the agent successfully learns the environmental dynamics by interacting with the instructor to perform correct actions to decrease the cognitive load. Figure 4(b) depicts the average energy consumed for each state in the form of a heatmap. Here, a darker color indicates a higher computational energy (i.e., higher cognitive load) required to perform visual recalling. Additionally, each cell in this heatmap corresponds to the same state in Figure 4(d). To derive the average energy for a state, we first record the total energy for that state, then we divide the total energy values by the number of visits to account for sensory noise.

The Q matrix after the end of 500 iterations is presented in Figure 4(c), which defines the behavior of the robot. In this figure, the states and actions (i.e., the next-states) were presented as rows and columns, respectively. We also use a heatmap to illustrate the Q matrix with a color bar in which the dark-blue cell displays the most valuable state-action pair. After learning the Q matrix, we select the best actions associated with the Q values for all states. For instance, according to the heatmap formation in Figure 4(c), we show the following state-action pair $(12, 11)$, which are illustrated with red dots in the same figure. If the robot visits state 12, its next-state will be 11 – i.e., the state with the highest Q value, which is marked with a red rectangle. We then assess whether the agent ends up circularly visiting some states or repeatedly visits a single state (i.e., singleton). We consider these states as the final discovered states by the agent and present these states with red rectangles in Figure 4(d). More importantly, we evaluate whether these final states are the states with low average energy values by comparing the final states with the same states in Figure 4(b).

In the following part, we present the results for all iterations grouped by the guiding strategy of the instructors. We note that the below descriptions are based on the same figures as above for all iterations. For the interested readers, these figures can be found in the repository of this paper (see Section VI).

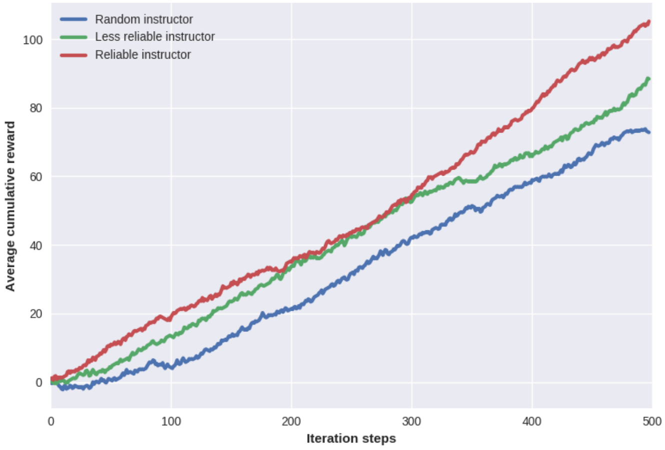


Fig. 5: Average cumulative reward for all instructors.

Figure 5 shows the average cumulative reward curves for five runs that were obtained after interacting with each instructor. Here the red, green, and blue lines refer to reliable, less reliable, and random instructors, respectively. In this figure, the reward curves for all instructors increase with the number of iteration steps. Yet, the increasing trend for the reliable instructor is higher than the others. This observation indicates that the interaction with the reliable instructor enables the agent to perform more correct actions (i.e., moving from a high-energy state to the lower one in order to minimize cognitive load) that generate positive reinforcements.

	Min	Max	Std	Avg	Interactions
Random instructor	40.0	92.0	18.3	72.8	1063
Less reliable instructor	74.0	100.0	10.5	88.4	1024
Reliable instructor	72.0	126.0	22.6	105.2	982

TABLE I: The statistics about all iterations for five runs.

Table I presents the statistics about five runs. In this table, the values in the columns indicate minimum, maximum, standard deviation, the average cumulative reward, and the total number of interactions with the instructors. In this table, the last column refers to the total number of interactions (i.e., decisions) made by the instructor according to the strategy followed, providing random, reliable, and less reliable decisions to the robot in order to perform an action. Based on the entries in this table, the interaction with a reliable instructor leads to obtaining the maximum reward for a single run and the highest average reward value for all runs. Moreover, the highest average cumulative reward value was achieved by the low number of interactions with the reliable instructor. We underline that this outcome indicates that interactions with a reliable instructor enable the agent to learn environmental dynamics (i.e., state-action-energy associations) to perform visual recalling by minimizing the cognitive load. Additionally, the reason why the standard deviation values are high in these runs might be explained by the external conditions of the experiment setup, such as the noise in the environment, lighting conditions, etc.

Since there is no predetermined final state to terminate the iteration in our setting, the robot should discover the final state or multiple final states by itself. We derived the discovered final states based on the final policy in the following way. After each run, we determined the highest Q values for all state-action pairs in the Q matrix to determine the final policy by following the greedy strategy to select the best action (i.e., the next state).

Run	Random Instructor	Less reliable Instructor	Reliable Instructor
	Final states	Final states	Final states
1	$9^*, 17, 10$	$1^*, 4$	$0^*, 12, 11^*, 13^*, 1^*$
2	$0^*, 16, 4, 17$	$1^*, 12, 14, 17$	$0^*, 7, 1^*, 10, 9^*, 8$
3	$0^*, 6, 7, 4, [3, 13^*, 17]$	$1^*, 17, 11^*, 3, 4, 7$	$1^*, 14, 11^*, 13^*, 17$
4	$0^*, 14, 6, [7]$	$0^*, 18, 8, 7, [17]$	$0^*, 6, 1^*, 16$
5	$0^*, 1^*, 6, 7, 2, 3$	$1^*, 12, 4, 8$	$0^*, 7, 1^*, 11^*, 4$

TABLE II: The discovered final states for all iteration steps with five runs. The state number with training patterns were shown with *.

The rows of Table II indicates the iteration runs, and the columns of the table show the discovered final states grouped by instructors. The interaction with the reliable instructor enables the agent to discover the final states populated with training patterns, which mostly lead to less energy to perform visual recalling. For instance, if we consider the discovered final states for all instructors in the fourth row of the Table II, the final states for the instructor with a random strategy will be $0^* \Rightarrow 14 \Rightarrow 6 \Rightarrow 0^*$, and a single state 7. Note that * indicates the states in which one of the stored patterns is located. Similarly, the interaction with the less reliable instructor will lead to finding either $0^* \Rightarrow 18 \Rightarrow 8 \Rightarrow 7 \Rightarrow 0^*$, and a single state 17. However, the final states with the reliable instructor ($0^* \Rightarrow 6 \Rightarrow 1^* \Rightarrow 16 \Rightarrow 0^*$) show different inclinations. First, the cycles obtained in the final policy contain two states that correspond to one of the stored patterns in the associative memory. Second, there are no singleton states that lead the robot to process the same visual pattern continuously. To show that interaction with the reliable instructor leads to less energy consumption, we derived the average energy cost for each final policy. We observe that the following energy-based ranking is formed: $Less\ reliable > Random > Reliable$. Based on the states in the final policy that generates state visitations and their corresponding average energy values, we emphasize that the interactions with the reliable instructor lead to performing visual recalling by low cognitive load.

Run	Random Instructor		Less reliable Instructor		Reliable Instructor	
	Correct	Wrong	Correct	Wrong	Correct	Wrong
1	16	4	17	3	16	4
2	18	2	17	3	18	2
3	17	3	14	6	19	1
4	18	2	19	1	19	1
5	17	3	19	1	19	1
%	86%	14%	86%	14%	91%	9%

TABLE III: The number of correct and wrong actions for all runs.

Table III was constructed to show the percentage of the correct and wrong actions taken by the robot after deriving the final policy based on the Q matrix at the end of each run, i.e., 500 iterations. The action of the robot was classified as correct if the robot moved from a high-energy state to a lower one. Otherwise, the action was considered as wrong. According to Table III entries, the interaction with the reliable instructor paves the way for taking more correct actions (91%) compared to the less reliable instructor and the instructor with random strategy (86%).

To further analyze the robot's behavior with different instructors, we depict the average temporal difference learning error. Here the error is defined as $td_{error} = R(s, s') + \gamma Q(s', a') - Q(s, a)$. As stated in Section III, R, $Q(s', a')$, and $Q(s, a)$ refer to the internal reward function, the value of the next action-state pair, and the value of the current state-action pair, respectively. We record the td_{error} value of each step during the experiment for each iteration and average them with the number of runs. Then, to capture the decreasing trend of the error curves, we performed a running average with a window size of 200.

Figure 6 illustrates the average temporal difference error curves for each instructor with the same color-code as we used in Figure 5 to indicate the instructor. The decreasing trends for each curve tends to approach zero –which indicates the learning progress towards an optimal Q function. The average temporal difference error of the reliable instructor

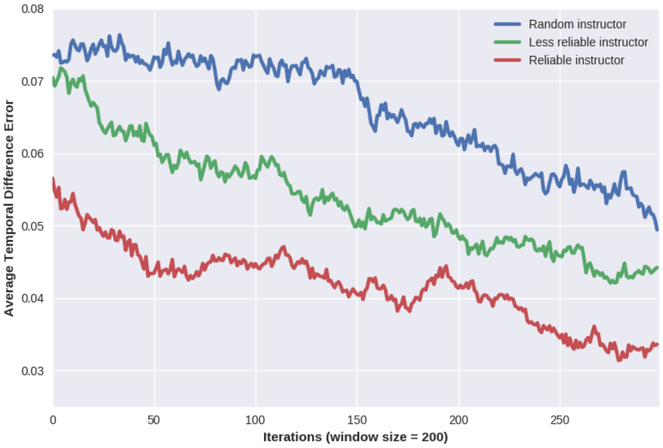


Fig. 6: Average temporal difference error curves for all instructors. The horizontal axis indicates the iterations with running average performed with a window size of 200.

yields the lowest error value and a better decreasing trend; that is, learning is improved compared with random and less reliable ones.

To sum up the presented results, we draw the following conclusions. First, the robot achieves the highest cumulative reward value with fewer interactions with the reliable instructor. Second, the interaction with the reliable instructor also enables the robot to perform the highest percentage of the correct actions with the final policy learned. In this way, the robot minimizes the required computational energy to perform a visual recalling task by moving its visual focus sequentially from the high-energy state to the lower one. Lastly, the reliable instructor also guides the robot to enhance its learning towards an optimum Q function to perform the sequential visual recall task efficiently.

Overall, the results indicate that the robot can differentiate the instructors' guiding strategies and select the reliable instructor as an interaction partner that can be trusted in case of a free choice for the next task. The performance metric to develop robot trust in an instructor could be the average reward, the number of correct actions, learning trend, etc. However, we suggest that the average cumulative reward is a more appropriate metric to assess robot trust in an interaction partner due to its interpretability on cost-aware decision throughout iterations and its formulation based on the association of the computational energy and visual recalling.

VI. REPRODUCIBILITY OF THE STUDY

We used a public repository [1] for resources (including data files, figures, visualization scripts) to reproduce the results. We note that the repository also hosts a video of the experiment demo.

[1] https://github.com/muratkirtay/ICDL2021/

VII. CONCLUSION

In this paper, we have demonstrated that the Pepper robot can develop trust in one of the online instructors by differentiating the guiding strategies – i.e., providing reliable, less reliable, or random guidance to the robot for performing a visual recalling task. Throughout interactions, the robot employs its auto-associative memory module to derive the cost of computation (i.e., cognitive load) then uses the cost values to generate an internal reward. Based on the average cumulative reward obtained at the end of interactions, the robot selects the instructor who reduces the cognitive load to perform the task. Overall, we show that the robot successfully differentiates the reliable instructor from the others in the case of free choice. We propose that this differentiation displayed by the robot is an indication of developing the robot's trust in an interaction partner.

As a continuation of this study, we would like to investigate the following research directions. First, in addition to the modules introduced in this study, we will leverage a multimodal learning architecture to enable the robot to process affective signals of the interaction partner (e.g., verbal cues and facial gestures) to examine the roles of these signals on the robot's emotional state and trust. Second, we will target integrating a co-representation mechanism in the same implementation to analyze the instructor's high-level cognitive skills (e.g., problem-solving, reasoning) on the robot trust [18]. Finally, we will design a robot-robot interaction experiment that the robot (e.g., the Pepper) will scaffold another robot (e.g., Nao) in a game-playing setting by utilizing the reliable instructor's strategies.

REFERENCES

[1] H. Atoyan, J.-R. Duquet, and J.-M. Robert, "Trust in new decision aid systems," in *Proceedings of the 18th Conference on l'Interaction Homme-Machine*, 2006, pp. 115–122.

[2] S. Shahrdar, L. Menezes, and M. Nojoumian, "A survey on trust in autonomous systems," in *Science and Information Conference*. Springer, 2018, pp. 368–386.

[3] Z. R. Khavas, S. R. Ahmadzadeh, and P. Robinette, "Modeling Trust in Human-Robot Interaction: A Survey," pp. 529–541, 2020.

[4] P. A. Hancock, D. R. Billings, and K. E. Schaefer, "Can you trust your robot?" *Ergonomics in Design*, vol. 19, no. 3, pp. 24–29, 2011.

[5] M. Kirtay, L. Vannucci, E. Falotico, E. Oztop, and C. Laschi, "Sequential decision making based on emergent emotion for a humanoid robot," *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pp. 1101–1106, 2016. [Online]. Available: http://ieeexplore.ieee.org/document/7803408/

[6] T. Sanders, K. E. Oleson, D. R. Billings, J. Y. Chen, and P. A. Hancock, "A model of human-robot trust: Theoretical model development," in *Proceedings of the human factors and ergonomics society annual meeting*, vol. 55, no. 1. SAGE Publications Sage CA: Los Angeles, CA, 2011, pp. 1432–1436.

[7] M. Novitzky, P. Robinette, M. R. Benjamin, D. K. Gleason, C. Fitzgerald, and H. Schmidt, "Preliminary interactions of human-robot trust, cognitive load, and robot intelligence levels in a competitive game," in *Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, ser. HRI '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 203–204. [Online]. Available: https://doi.org/10.1145/3173386.3177000

[8] M. I. Ahmad, J. Bernotat, K. Lohan, and F. Eyssel, "Trust and cognitive load during human-robot interaction," *arXiv*, 2019.

[9] F. Correia, P. Alves-Oliveira, N. Maia, T. Ribeiro, S. Petisca, F. S. Melo, and A. Paiva, "Just follow the suit! trust in human-robot interactions during card game playing," in *2016 25th IEEE international symposium on robot and human interactive communication (RO-MAN)*. IEEE, 2016, pp. 507–512.

[10] M. Patacchiola and A. Cangelosi, "A developmental bayesian model of trust in artificial cognitive systems," in *2016 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, 2016, pp. 117–123.

[11] M. Patacchiola and A. Cangelosi, "A developmental cognitive architecture for trust and theory of mind in humanoid robots," *IEEE Transactions on Cybernetics*, 2020.

[12] S. Vinanzi, M. Patacchiola, A. Chella, and A. Cangelosi, "Would a robot trust you? developmental robotics model of trust and theory of mind," *Philosophical Transactions of the Royal Society B*, vol. 374, no. 1771, p. 20180032, 2019.

[13] T. Chaminade, E. Oztop, G. Cheng, and M. Kawato, "From self-observation to imitation: Visuomotor association on a robotic hand," *Brain research bulletin*, vol. 75, no. 6, pp. 775–784, 2008.

[14] M. Kirtay and E. Oztop, "Emergent emotion via neural computational energy conservation on a humanoid robot," *IEEE-RAS International Conference on Humanoid Robots*, vol. 2015-February, no. February, pp. 450–455, 2015.

[15] M. Kirtay, L. Vannucci, U. Albanese, C. Laschi, E. Oztop, and E. Falotico, "Emotion as an emergent phenomenon of the neurocomputational energy regulation mechanism of a cognitive agent in a decision-making task," *Adaptive Behavior*, p. 1059712319880649, 2019.

[16] J. Hertz, A. Krogh, and R. G. Palmer, *Introduction to the Theory of Neural Computation*. USA: Addison-Wesley Longman Publishing Co., Inc., 1991.

[17] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press Cambridge, 1998, vol. 1, no. 1.

[18] M. Kirtay, O. A. Wudarczyk, D. Pischedda, A. K. Kuhlen, R. A. Rahman, J. D. Haynes, and V. V. Hafner, "Modeling robot co-representation: state-of-the-art, open issues, and predictive learning as a possible framework," in *2020 Joint IEEE 10th International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, Oct 2020, pp. 1–8.