

Neuromodulation, Attention and Localization Using a Novel Android™ Robotic Platform

Nicolas Oros, Jeffrey L. Krichmar

Abstract—We present a novel neural architecture based on neuromodulated attentional pathways. The neural network controlled a robot, which had to perform a reversal learning task based on GPS locations. We developed a robotic platform that leverages smartphones technology. The behavior of the robot was entirely driven by a neural network that ran on an Android phone, which handled sensor input from the phone and controlled the motor and servo of the robot. The robot managed to perform the task successfully by increasing attention to relevant locations and decreasing attention to irrelevant ones.

Index Terms—Attention, Android phones, Neuromodulation, Robot.

I. INTRODUCTION

THE ability to fluidly divert attentional resources toward relevant information is critical for successful behavior in animals and may be applicable for the control of autonomous behavior. Humans, monkeys and rodents, possess a network of cholinergic neurons throughout the basal forebrain that project to the hippocampus and the cortex. Modulating the release of acetylcholine (ACh) in the respective target structures is likely responsible for the ability to refine attention for optimal performance, resulting in ignoring irrelevant cues in the environment and attending to important cues. Experiments conducted by Chiba [1] and Baxter [2] showed that the basal forebrain, the main source of acetylcholine, has specific and separate pathways for decrementing and incrementing attentional effort. Whereas ACh projections from the medial septum/vertical limb of the diagonal band (MS/VDB) to the hippocampus and medial frontal cortex are crucial to reduce attention to irrelevant stimuli, ACh projections from the substantia innominata/nucleus basalis region (SI/nBM) to the neocortex and amygdala are necessary to increase attention to relevant stimuli.

Understanding the principles of neuromodulation, such as acetylcholine, dopamine, norepinephrine and serotonin functionality, may inform control and action selection algorithms for autonomous robots. Recently, Krichmar introduced an action selection mechanism for robots based on principles of the neuromodulatory system [3, 4]. The neural controller, which has been tested on autonomous robots,

Manuscript received October 11, 2012. This work was supported in part by the NSF under Grant IIS-0910710.

N. Oros and J. L. Krichmar are with the Department of Cognitive Sciences, University of California, Irvine, Irvine, CA 92697-5100 USA, {noros; jkrichma}@uci.edu.

demonstrated valuable features such as, fluid switching of behavior, gating in important sensory events, and separating signal from noise.

In this paper, we present a neurobiologically inspired neural model that implements principles of neuromodulation, and in particular focuses on how ACh modulates the ability to increase attention to relevant stimuli and decrease attention to distractions. In addition, we introduce a novel robot that combines Android smartphone technology with off-the-shelf remote control (R/C) cars. The computing power and sensing capabilities of these smartphones affords an inexpensive yet highly capable robotic platform.

II. MOTIVATION

Attentional set-shifting experiments, such as the Wisconsin Card Sorting Task and reversal learning, are often performed to study the ability to switch between behavioral strategies. In reversal learning, the subject must initially learn a stimulus-reward association, and then demonstrate the ability to switch strategies when the experimenter introduces a new stimulus-reward association. A common behavior observed in animals when a reversal is introduced is perseverative behavior, which consists of behavioral responses to the previously reinforced stimulus [5-7].

We investigated the influences of ACh on both the incremental and decremental pathways for regulating attention and learning during a reversal task. The role of ACh in regulating attention has been investigated by several computational models [8-10]. However, there is little experimental evidence and few modeling studies that have investigated how ACh can decrement or increment attention in a reversal task. We previously developed a neural simulation to provide insight into how acetylcholine can decrement or increment attention using two distinct pathways, and how dopamine and noradrenaline influence these pathways [11]. This model exhibited behavioral effects such as latent inhibition, persisting behavior, habituation to stimuli, and increased learning when facing novel or unexpected stimuli and reward associations.

In the present paper, our goal was to use a simplified version of this model in order to investigate how the incremental and decremental pathway could guide the behavior of a robot performing a reversal learning task.

III. ANDROID ROBOTIC PLATFORM

The computational power of handheld devices, such as mobile phones and tablets, increases every year at a

remarkable rate. Even though smartphones have compact form factors, they are currently equipped with powerful dual-core processors and graphical processing units, video cameras, location providers (GPS, Wi-Fi, Cell-ID), and a multitude of sensors such as acceleration and orientation sensors. They also have an impressive suite of connectivity options (USB, Bluetooth, Wi-Fi, 3G, 4G), are powered by small, but long-lasting batteries, run modern operating systems (OS), and are reasonably priced. For software development, many smartphone OS's provide a Software Development Kit (SDK) that enables programmers to create applications relatively easily. For these reasons, we believe that Android phones are ideal candidates for onboard computing in autonomous robots. By using a smartphone as an onboard computer, the size of a robot can be kept relatively small and still have great computational features. Its cost can also be minimal since the phone itself can handle computation and sensing, and the actuating platform can be kept relatively cheap and small by using an off-the-shelf remote control (R/C) car as a robot chassis. These R/C cars typically provide a speed controller to regulate forward velocity and a servomotor (servo) for steering.

Therefore, we created the Android Robotic Platform characterized by its low cost, robustness, flexibility, modularity and facility to use. In order to control a R/C car from an Android phone, we used the IOIO (pronounced yo-yo) board (<http://www.sparkfun.com/products/10748>) to link the phone to the car's motor and servo. The IOIO serves as a communication bridge between the Android Smartphone and the device, using USB or Bluetooth. In our case, commands (PWM signals) are sent from the Smartphone to the R/C car's speed controller and steering servo through the IOIO, and the IOIO sends the values of infrared sensors (IR) mounted on the robot, which are necessary for obstacle avoidance, to the Smartphone. The platform consists of the chassis of a R/C car (HP Vertex RTR 4WD 1/10 scale), a base (perforated steel) that supports the IOIO, IR sensors, a phone holder (polystyrene foam) and an Android phone (HTC Incredible 1). The perforated base allows roboticists to add, remove or relocate sensors easily on the robot. We named the first robot created using this platform "Le CARL" (Fig. 1, left). For more details on the construction of our robotic platform, see [12].

IV. METHODS

A. Task

We used our Android controlled robot to perform a task involving reversal learning. The robot had an energy level that decreased over time, with a value comprised between zero and one. When the robot found resources (reward) in the environment, its energy level increased. The robot had to learn to go to the reward location when its energy level was low. The experiment was conducted outdoors on an open grass field where two GPS locations were chosen (L_1 and L_2) and only one location contained resources (see Fig. 1, right). A location (L_1 or L_2) was defined as a circular area with a radius of 5m. The center of L_1 was 15m away from the center of L_2 . The robot would "consume" resources when it was inside the radius of a location with resources. Once the agent

"consumed" all the resources present at one location, a reversal was introduced by placing the resources at the other location. We chose a value for the amount of resources such that it would fully replenish the robot's energy by visiting the location three times.

The agent's basic behavior was to explore its environment. The resources were initially located at L_1 . The robot was initially placed at the non-rewarding location L_2 facing in the opposite direction of the rewarding location L_1 . This was to ensure that the robot would learn about L_2 before learning about L_1 , allowing reversal learning once the resources of L_1 were consumed. The experiment consisted of ten trials and the locations were selected at different places on the open field for each trial.



Fig. 1. Left - "Le CARL" robot constructed from the chassis of a R/C car. A base consisting of a sheet of perforated steel was mounted onto the chassis. Four infrared sensors, a phone holder, an Android smartphone, and the IOIO were mounted onto the base. Right - open grass field with both locations L_1 and L_2 , where the experiment was conducted.

B. Neural Architecture

A neural network (NN) running on the Android phone controlled the robot and received inputs from the phone such as the GPS location, compass reading (azimuth) and the values of the IR sensors connected to the IOIO. The network processed the sensory information in order to learn where the reward was located and select a location to attend to, and outputted the signals controlling the motor and servo of the robot.

The neural network driving the behavior of the robot was composed of three main groups (see Fig. 2): sensory input, action selection and motor output. Every neuron in these groups were modeled as mean firing rate neurons defined by:

$$s_i(t) = \frac{1}{1 + \exp(G_i(\theta_i - I_i(t)))} \quad (1)$$

where t is the current time step, s_i is the activation level of neuron i , G_i is the neuronal gain, θ_i is a threshold and I_i is the total synaptic input. For each neuron, the default value of G_i was set to 10 and θ_i to 0.5 in order to have neural activities ranging from zero to one when stimulated by a single neuron.

The synaptic input of a neuron was based on pre-synaptic neural activity, and the connection strength of the synapse:

$$I_i(t) = \sum_j w_{ij}(t-1)s_j(t-1) \quad (2)$$

where w_{ij} is the synaptic weight from neuron j to neuron i . The default value of the weights for all non-plastic projections was set to 1 for excitatory and -1 for inhibitory connections.

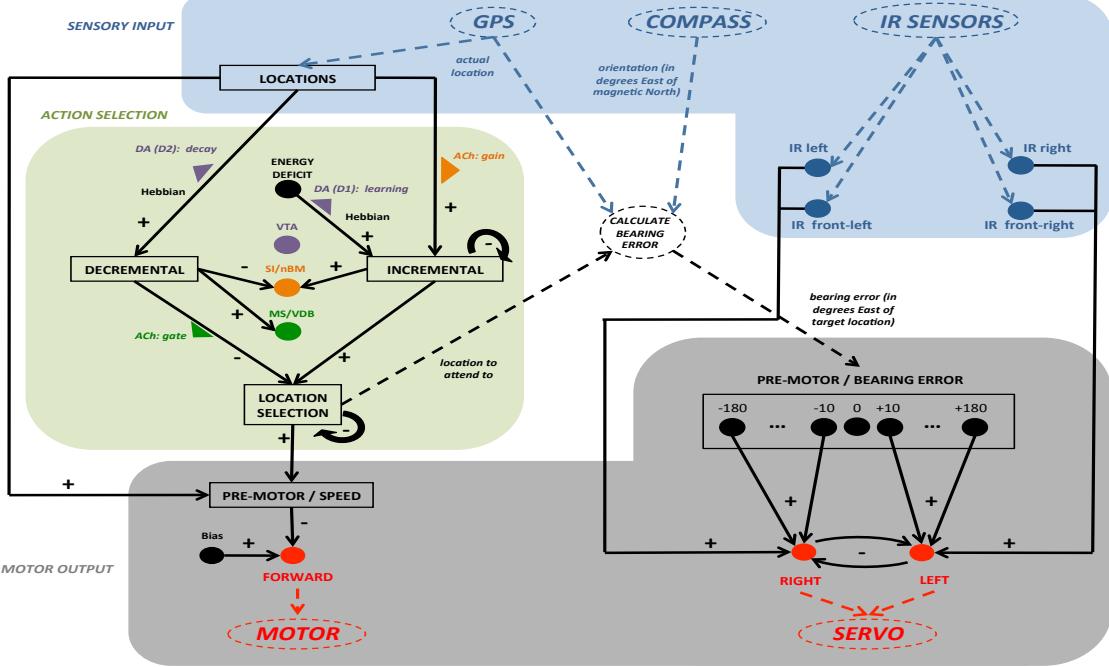


Fig. 2. Neural architecture driving the behavior of the robot. It was composed of three main groups: sensory input, action selection and motor output. Solid line items represent neural implementation (neurons and connections).

1) Sensory input

The robot received its current location in GPS coordinates (latitude and longitude) at approximately 1Hz. The magnetic compass of the phone was used to give the robot's orientation (azimuth) in degrees East of magnetic North at approximately 10Hz. The locations were encoded neurally by a locations area that consisted of two neurons, one for each location (L_1 and L_2). The neural activity of this area was based on the distances between the actual location and the recorded ones (L_1 and L_2). The input of these neurons was calculated using the following equation:

$$I_i(t) = 1 - \frac{d_i}{K^d} \quad (3)$$

where d_i is the distance in meters between the actual GPS location and the location L_i , and K^d is a constant set to 10. The activity of the Locations neurons was then calculated using Equation 1 with $G_i = 50$ to model a Heaviside step function causing the neurons to be activated and saturate only when the robot was within 5m of the center of the location. The distance d_i was calculated using the Android API. Four infrared sensors were connected to the IOIO and were used to detect obstacles. Four neurons were used to encode the value of these IR sensors. The activity of these neurons was calculated using Equation 1 with the input I being the IR sensor's analog value scaled between 0 and 1, $G_i = 20$ and $\theta_i = 0.3$. These values were chosen to filter out low values (mostly noise) and cause the neurons to saturate for an input value I of ~ 0.6 .

2) Action selection

The action selection group was based on the known functional neuroanatomy for attentional pathways. It consisted of a decremental and an incremental attention area, and a location selection area. These areas had two neurons, one for each location (L_1 and L_2). The decremental area roughly corresponded to the hippocampus and anterior cingulate cortex

(septo-cingulate pathway), and the incremental area roughly corresponded to the amygdala combined with the OFC. We also modeled three neuromodulatory areas (MS/VDB, SI/nBM, and the ventral tegmental area - VTA) consisting of one neuron each. Finally, we included a neuron representing the energy deficit of the robot ($deficit = 1 - energy$). The decremental, incremental and location selection areas received one-to-one connections. The incremental area also received one-to-all connections from the energy deficit neuron. The incremental and location selection areas had lateral inhibitory connections (with $w_{ij} = -0.8$ causing a strong inhibition). The MS/VDB and SI/nBM received all-to-one connections from the decremental and incremental (for SI only) areas. The main function of the action selection group was to learn that a location was predictive of a reward, and to choose a location to attend to, causing the robot to stop at a novel location, or to go back to the reward location when the robot's energy was low.

The role of dopamine in reinforcement learning has been clearly established over the last decades [13]. Experiments have highlighted a role for D1 dopamine receptors in synaptic potentiation [14], and for D2 dopamine receptors in synaptic depression [15]. Therefore, we included Hebbian learning modulated by dopamine in the model (see purple neuron and triangle in Fig. 2). The amount of dopamine emitted from the VTA ($DA(t)$) was calculated based on the reward received and the energy level of the robot:

$$\sigma(t) = reward(t) - energy(t) \quad (4)$$

$$DA(t) = \begin{cases} 0 & \text{if } \sigma(t) < 0 \\ 5\sigma(t) & \text{otherwise} \end{cases} \quad (5)$$

where $reward(t)$ represents the presence (0 or 1) of a reward/resources at a particular location, and $energy(t)$ is the energy level of the robot. The value of $reward(t)$ was set to

one when the robot was inside the radius of a location containing resources.

The weights from the energy deficit neuron to the incremental area were plastic and modulated by dopamine:

$$\Delta w_{ij}(t) = \epsilon(w_{ij}(0) - w_{ij}(t-1)) + DA(t) \delta s_i(t)s_j(t-1) \quad (6)$$

The maximum value of w_{ij} was capped at one, its initial value was 0.1, and the decay and learning rates were set to $\epsilon=0.001$, and $\delta=0.8$, respectively. These values were chosen to allow fast learning in the presence of dopamine, and slow forgetting. The effect of dopaminergic D1 receptors was modeled here by having DA levels increase synaptic weights. Reinforcement learning modulated by dopamine caused the robot to remember the location where the reward was located in order to go back to it when its energy level decreased.

In our model, connections from the locations area to the decremental area were also subject to Hebbian learning modulated by dopamine where the change of synaptic strength was defined by:

$$\Delta w_{ij}(t) = DA(t)s_i(t)\epsilon(w_{ij}(0) - w_{ij}(t-1)) + \delta s_i(t)s_j(t-1) \quad (7)$$

where $\Delta w_{ij}(t)$ is the change of weight, ϵ is the decay rate, δ is the learning rate. The maximum value of w_{ij} was capped at one, its initial value was 0.01, and the decay and learning rates were set to $\epsilon=0.8$, and $\delta=0.1$, respectively. These values were chosen to allow fast learning, but faster forgetting in the presence of dopamine. The effect of dopaminergic D2 receptors was modeled here by having DA levels cause synaptic depression through the decay term. Functionally, the decremental area learned about locations that were or became irrelevant (no reward received) in order to decrease the robot's attention to these locations. However, the decremental area would unlearn, or not learn, about locations when the robot received a reward.

The neural activity of the MS/VDB was calculated using Equation 1 with $G_i=40$ and $\theta_i=0.2$ causing the MS to rapidly saturate when stimulated. The cholinergic neuron of the MS/VDB gated the inhibitory signal coming from the decremental area to the location selection area using the following equations:

$$I_i^{DEC}(t) = \sum_j s^{MS}(t-1)w_{ij}(t-1)s_j(t-1) \quad (8)$$

where $I_i^{DEC}(t)$ is the input from the decremental to the location selection area, s^{MS} is the activity of the MS/VDB cholinergic neuron. Functionally, neuromodulation from MS/VDB activated the decremental attentional pathway resulting in the inhibition of the location selection area (see green neuron and triangle in Fig. 2).

The neural activity of the SI/nBM neuron modulating the synaptic input of the incremental area was given by:

$$I_i(t) = \sum_j (1 + s^{SI}(t-1))w_{ij}(t-1)s_j(t-1) \quad (9)$$

where s^{SI} is the activity of the cholinergic neuron in the SI/nBM. Functionally, ACh increased the gain on target neurons [16, 17] (see orange neuron and triangle in Fig. 2).

3) Motor output

The motor output group consisted of a pre-motor/speed area composed of one neuron for each location (L_1 and L_2), a pre-motor/bearing error area composed of 37 neurons (10 degree resolution), and three motor neurons: one to move forward,

one to turn right and one to turn left.

The neural activity of the pre-motor/speed area was calculated using Equation 1 with $G_i=30$ and a high threshold $\theta_i=1.2$. This area received inputs from both the locations and the location selection areas with one-to-one connections. Due to the high threshold and gain values, a neuron in the pre-motor/speed area only fired when both input neurons encoding the same location fired at the same time. This would cause an inhibition of the forward motor neuron and stop the robot. Therefore, the robot would only stop when it reached the location it wanted to attend to (output of the location selection area). The neural activity of the motor neuron (forward) was calculated using the following equation:

$$s_i(t) = \rho_i s_i(t-1) + (1 - \rho_i) \left(\frac{1}{1 + \exp(G_i(\theta_i - I_i(t)))} \right) \quad (10)$$

where ρ is the persistence of neural activity.

We chose a high value for ρ (0.9) that would cause the activity of the motor neuron to increase and decrease gradually so the robot would not change its speed too abruptly. The forward motor neuron received a bias input causing the robot to move forward by default. The activity of this neuron was then mapped to the pulse width of the PWM signal controlling the robot's motor using the following equation:

$$pwm^{motor}(t) = 1500 + (K^{motor} s^{forward}) \quad (11)$$

where K^{motor} is a constant set to 100 to limit the maximum pulse width of the PWM signal to 1600, and $s^{forward}$ is the activity of the forward neuron. We limited the value of the pulse width in order to have a maximum speed of approximately 8mph.

The steering of the robot was based on the activity of the right and left motor neurons that received inputs from the pre-motor/bearing error area. The activity of this area corresponded to the bearing error between the orientation of the robot and the bearing to the target location, and was calculated in the following steps. If the activity of a neuron in the location selection area reached 0.5, the bearing (ϕ^{GPS}) from the actual GPS location to the chosen GPS location was calculated using a function provided by the Android API, which outputted an angle in degrees East of the true North. We then read the azimuth from the compass (angle East of magnetic North), and had to correct this value in order to have the orientation of the robot East of the true North (ϕ^{robot}). This was performed by adding to the angle the declination of the magnetic field from the true North at the actual GPS location (using the Android API), and the orientation of the phone on the robot (+90 degrees, see Fig. 1, left). We could then calculate the bearing error (ϕ^{error}):

$$\phi^{error}(t) = \phi^{robot}(t) - \phi^{GPS}(t) \quad (12)$$

$$\phi^{error}(t) = \begin{cases} \phi^{error}(t) - 360 & \text{if } > 180 \\ \phi^{error}(t) + 360 & \text{if } < -180 \end{cases} \quad (13)$$

The value of ϕ^{error} was then mapped into the neural activity of the pre-motor/bearing error area with a resolution of 10 degrees per neuron (see Fig. 2). The higher the value of ϕ^{error} was, the more neurons would fire. For example, a bearing

error of +40 degrees would mean that the robot was 40 degrees East of the target location, and would cause the activation of four pre-motor neurons (+10, +20, +30 and +40). All the neurons encoding a negative ϕ^{error} projected to the right motor neuron, whereas the ones encoding a positive value projected to the left motor neuron. If the activity of any neuron in the location selection area did not reach 0.5, ϕ^{error} was set to zero so the robot would move forward by default, except when the agent was more than 5m away from the closest location, where the value of ϕ^{GPS} was set using a location randomly selected (L_1 or L_2).

The neural activity of each motor neurons (left and right) was calculated using Equation 1 with $G_i=2$ and $\theta_i = 2.5$. These values were chosen to cause a saturation of the motor neurons only when stimulated by four pre-motor neurons or more. The two motor neurons inhibited strongly each other ($w_{ij} = -20$), and received strong projections from the neurons encoding the values of the IR sensors ($w_{ij} = 20$). In order to implement a simple obstacle avoidance mechanism, the left IR neurons projected to the right motor neuron whereas the right IR neurons projected to the left motor neuron. The activity of the left and right motor neurons was mapped into the pulse width of the PWM signal controlling the robot's servo in order to turn the front wheels using the following equation:

$$pwm^{servo}(t) = 1500 + K^{servo} \left(s^{right} - s^{left} \right) \quad (14)$$

where K^{servo} is a constant set to 500, s^{right} is the activity of the right motor neuron, and s^{left} is the activity of the left motor neuron. Therefore, the range of values used for $pwm^{servo}(t)$ was between 1000 and 2000.

V. RESULTS

A. Behavior

The robot successfully performed the task in 8.5 minutes on average (Fig. 3). Experimental trials are described below and can be seen on a video online [12]. We set up the experiment so that the robot always started at the second location L_2 . The robot initially learned that no resources were present at this location and started to move. If the robot went back to this location again, it would not stop since it knew that L_2 did not contain a reward. During this time, the robot's energy level kept decreasing. Once the robot found location L_1 where the reward was located, it stopped and stayed still until its energy level was fully replenished. During this time, the robot learned to associate resources with the location L_1 . Once its energy level was full, the robot started to move again and explore its environment. When its energy level was low, the robot would go back to the location L_1 associated with the reward. The robot consumed all the resources present at L_1 , by visiting L_1 three times, in 3.1 minutes on average (Fig. 3). A reversal was then introduced by placing resources at location L_2 . However, the robot persisted in going back to L_1 for ~2 minutes (Fig. 3) until it finally switched back to an explorative behavior. The robot then found and learned that the resources were now at the second location L_2 . As before, when its energy level was low, the robot would go back to the location L_2 . The robot consumed all the resources present at L_2 in 3.3 minutes on average (Fig. 3). These results show that the robot managed to

perform the task in a short amount of time during which it initially learned a stimulus-reward association, and then demonstrated the ability to switch strategy when a reversal was introduced. The robot also exhibited perseverative behavior in accordance with the behavior observed in rats [6, 7] and monkeys [5] performing a reversal learning task.

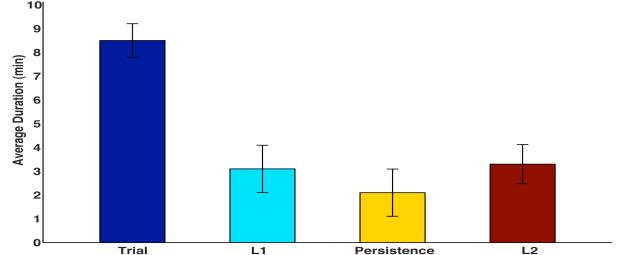


Fig. 3. Average duration in minutes necessary for the robot to perform the task, and consume all the resources contained at locations L_1 and L_2 . The duration of the robot's persisting behavior after a reversal is also shown. The mean and standard deviation (error bars) were calculated over ten trials. The values were normally distributed.

B. Neural Activity

We recorded the neural activity of the robot while performing the task. Fig. 4 shows the activity of the NN in a representative trial when the robot was low on energy and was going back towards the learned location that contained the reward. Looking at Fig. 4, we can see that the energy deficit of the robot was high during the first 11s, causing the neuron 1 in the incremental area to fire. Consequently, the activity of neuron 1 of the location selection area was above 0.5 meaning that location L_1 was selected. The bearing error was then calculated, which caused the left and right neuron to fire accordingly, setting the value of the PWM servo signal allowing the robot to turn towards the location L_1 . During this time, the robot was in motion as shown by the activity of the forward neuron and the value of the PWM motor signal. Once the robot reached location L_1 , the robot started to receive a reward, increasing the VTA dopamine signal (DA) and slowly replenishing the robot's energy level. Neuron 1 of the locations area was highly active, causing a strong excitation of the incremental neuron. Neuron 1 of the decremental area also was active but the strong depressing effect of DA on the synapses between the locations and decremental area caused it to stop rapidly. With both neurons 1 of the locations and location selection area firing, the pre-motor/speed neuron saturated, causing a strong inhibition on the forward neuron and thereby, stopping the robot. The robot could then consume the resources and its energy deficit decreased as well as the amount of DA. With no more dopamine, the decremental area started to learn again and its activity increased after 40s (Fig. 4). The MS was then stimulated and gated the inhibitory signal to the location selection area that stopped firing. The pre-motor/speed area did not receive enough stimulation and therefore ceased activity. With no inhibition coming from the pre-motor area, the activity of the forward neuron kept increasing and the robot started moving again. The energy of the robot was replenished and it was now able to continue exploring its environment.

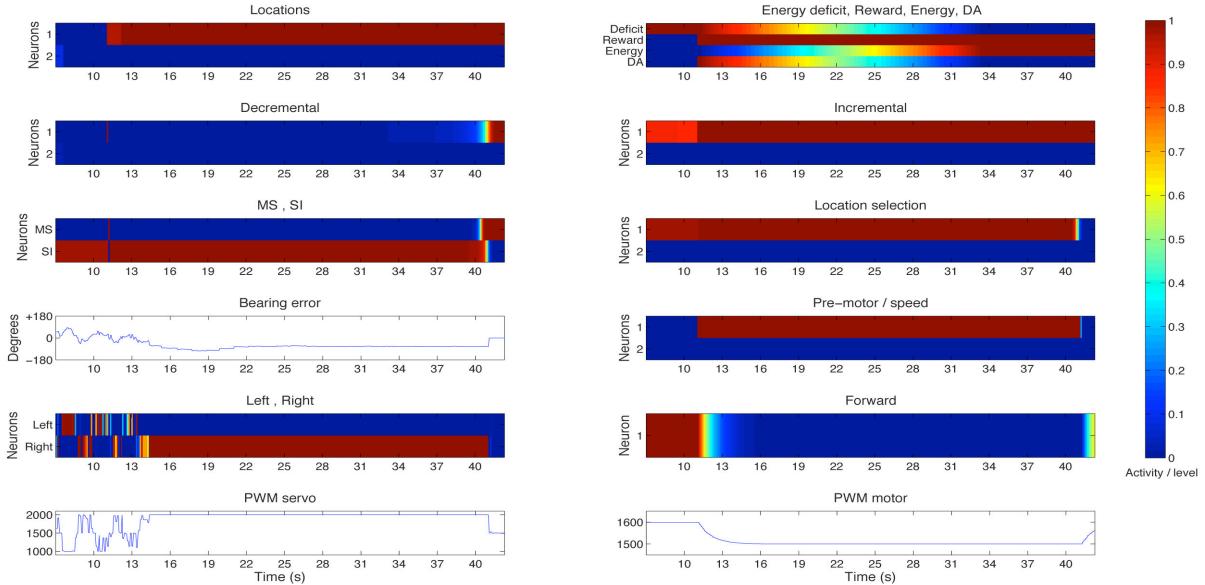


Fig. 4. Activity of the neural network when the robot was low on energy and was going back towards the location associated with a reward.

VI. DISCUSSION

We presented a neural architecture based on neuromodulated attentional pathways that controlled an autonomous robot carrying out a reversal task. The decremental pathway, modulated by ACh from the MS and DA from the VTA, learned to decrease attention to irrelevant locations. Whereas the incremental pathway, modulated by ACh from the SI and DA from the VTA, increased attention to relevant locations and learned which location contained a reward. The behavior of the robot was entirely driven by its neural network and the robot managed to perform a reversal learning task successfully by increasing its attention to relevant location and decrease its attention to irrelevant ones. We observed that the robot was always able to find the two locations even though the accuracy of the GPS was quite low and could vary over time. We also presented a novel robotic platform that can utilize the numerous features of Android phones.

Using a robotic platform composed of an Android phone, an IOIO board and a R/C car is a promising approach for inexpensive robotics research and for education. Furthermore, the incremental and decremental attention pathways presented in this paper can be used for practical applications in search and exploration domains.

ACKNOWLEDGMENT

The authors would like to thank L. Bucci for assistance during the construction of the Android Robotic Platform, as well as A. Chiba and D. Nitz for many useful discussions.

REFERENCES

- [1] A. A. Chiba, *et al.*, "Basal forebrain cholinergic lesions disrupt increments but not decrements in conditioned stimulus processing," *J Neurosci*, vol. 15, pp. 7315-22, Nov 1995.
- [2] M. G. Baxter, *et al.*, "Disruption of decrements in conditioned stimulus processing by selective removal of hippocampal cholinergic input," *Journal of Neuroscience*, vol. 17, pp. 5230-5236, Jul 1 1997.
- [3] B. R. Cox and J. L. Krichmar, "Neuromodulation as a Robot Controller: A Brain Inspired Design Strategy for Controlling Autonomous Robots," *IEEE Robotics & Automation Magazine*, vol. 16, pp. 72-80, 2009.
- [4] J. L. Krichmar, "A biologically inspired action selection algorithm based on principles of neuromodulation," in *The 2012 International Joint Conference on Neural Networks (IJCNN)*, 2012, pp. 1-8.
- [5] B. Jones and M. Mishkin, "Limbic lesions and the problem of stimulus-reinforcement associations," *Exp Neurol*, vol. 36, pp. 362-77, Aug 1972.
- [6] V. Boulougouris, *et al.*, "Dopamine D2/D3 receptor agonist quinpirole impairs spatial reversal learning in rats: investigation of D3 receptor involvement in persistent behavior," *Psychopharmacology (Berl)*, vol. 202, pp. 611-20, Mar 2009.
- [7] S. Ghods-Sharifi, *et al.*, "Differential effects of inactivation of the orbitofrontal cortex on strategy set-shifting and reversal learning," *Neurobiol Learn Mem*, vol. 89, pp. 567-73, May 2008.
- [8] A. J. Yu and P. Dayan, "Uncertainty, neuromodulation, and attention," *Neuron*, vol. 46, pp. 681-92, May 19 2005.
- [9] M. Usher and E. J. Davelaar, "Neuromodulation of decision and response selection," *Neural Netw*, vol. 15, pp. 635-45, Jun-Jul 2002.
- [10] M. C. Avery, *et al.*, "Simulation of cholinergic and noradrenergic modulation of behavior in uncertain environments," *Front Comput Neurosci*, vol. 6, p. 5, 2012.
- [11] N. Oros, *et al.*, "Simulation of the neuromodulatory pathways responsible for incrementing and decrementing attention during reversal learning and set-shifting," in *Proceeding of the Society for Neuroscience (SFN 2011)*, Washington, D.C., 2011.
- [12] N. Oros. (2012). *Android Robotic Platform*. Available: http://www.cogsci.uci.edu/~noros/android_robotic_platform.html
- [13] W. Schultz, *et al.*, "A neural substrate of prediction and reward," *Science*, vol. 275, pp. 1593-9, Mar 14 1997.
- [14] S. Li, *et al.*, "Dopamine-dependent facilitation of LTP induction in hippocampal CA1 by exposure to spatial novelty," *Nat Neurosci*, vol. 6, pp. 526-31, May 2003.
- [15] B. Pan, *et al.*, "D2 dopamine receptor activation facilitates endocannabinoid-mediated long-term synaptic depression of GABAergic synaptic transmission in midbrain dopamine neurons via cAMP-protein kinase A signaling," *J Neurosci*, vol. 28, pp. 14018-30, Dec 24 2008.
- [16] A. A. Disney, *et al.*, "Gain modulation by nicotine in macaque v1," *Neuron*, vol. 56, pp. 701-13, Nov 21 2007.
- [17] M. E. Hasselmo and J. McGaughy, "High acetylcholine levels set circuit dynamics for attention and encoding and low acetylcholine levels set dynamics for consolidation," *Prog Brain Res*, vol. 145, pp. 207-31, 2004.