Developmental Robot Learning through Multiple Teaching Modes

Arden Knoll and Jacob Honer

MICHIGAN STATE UNIVERSITY GENISAMA

Building robust autonomous systems through a traditional engineering approach is challenging - if not impossible. Recently, machine learning has been a solution to this difficulty, as it allows robots to learn the complex dynamics of their environment and actions through either direct supervision or reinforcement. Both learning modes have their drawbacks; motor supervision is tedious for the trainer and reinforcement takes increased time. Reinforcement learning promises to make the training process easier and allows for better generalization. The pairing of the two may create a more efficient training process. Here, in addition to the above learning modes, we introduce an unsupervised learning mode, or practice mode, where the system generates its own actions and learns by observing the effects of said actions. Our robot learning framework uses multiple teaching modes (motor-supervision, reinforcement, and practice), and we demonstrate its capabilities through a series of experiments involving navigation tasks using a single stereo-camera. Unlike most neural networks, Developmental Networks (DNs), used in this work, do not rely on gradient descent-based learning algorithms, and are therefore able to learn optimally, through real-time interactions with their environment, across a single lifetime. In this work we analyze and discuss the effects of each training mode on a DN's performance of a navigation task.

INTRODUCTION

Direct supervision is the most efficient training method because it provides correct labels to the system. However, labeling is difficult for a human to do in real-time and becomes impossible as the number of motors supervised increases. Therefore, we need new real-time learning methods that are less tedious for a human trainer to manage. Here, we explore training a DN with the direct supervision method that we used in prior work, in addition to a reinforcement method and a practice method, which attempts to address the above concerns.

Ideally, we would train our system frame by frame over a single lifetime. This is so that the system can learn directly how its actions affect both itself and its environment. This process lends itself to ideal training being on real, physical systems. In order to quantify our new training methods, however, we opted for batch training so that we would have more control over our training, and repeatable and observable changes. Our training is still frame incremental.

Reinforcement Mode

The reinforcement method proposed follows a reward and punishment model. This model was inspired by the human body's secretion of either dopamine or serotonin as a biological reinforcement in response to reinforcement in the real-world. When the DN produces a good predication, we reinforce with dopamine, and when it produces a bad predication, we punish with serotonin. The effects of this dopamine and serotonin effect the motor and hidden neurons of the DN differently.

For motor neurons, the effects are to weaken or strengthen the pre-action potential of the target motor neurons, which determines their likelihood to fire. Dopamine increases the pre-action potential, and inversely, serotonin decreases the pre-action potential of motor neurons.

$$z_i = r_{iu} \gamma\,(1 - \alpha\, w_{ip} + \beta\, w_{is}) \qquad (1)$$

Equation 1 shows how the "unbiased" pre-action potential, r_{iu}, is the scaled by the serotonin level, w_{ip}, and the dopamine level, w_{is}, to become the pre-action potential, z_i. The γ, α, and β, and are constants, set to 1.0, 0.9, and 0.3, respectively. These values can be optimized experimentally.

For the hidden neurons, dopamine and serotonin increases the learning rate, w_2, of the firing neuron, and decreases the retention rate, w_1, so that neurons can better memorize reinforcement events.

$$w_2 = \min\left\{\left(1 + w_{is} + w_{ip}\right)\frac{1 + \mu(a_i)}{a_i}, 1\right\}; w_1 = 1 - w_2 \qquad (2)$$

Equation 2 shows this relationship, where $\mu(a_i)$ is the amnesic function, and a_i is the neuron's age [1].

In summary, a punished event would weaken the pre-action potential of the firing motor neurons, and the hidden neurons, with an increased learning rate, would more quickly memorize these decreased top-down motor weights. Likewise, with a rewarded event, the reverse would occur with the motor neurons, where the pre-action potentials would be increased, but these reinforced top-down motor weights would be more quickly learned by the hidden neurons because of the increased learning rate.

Practice Mode

Our proposed practice method allows to DN to run freely and update using its own actions. This allows the DN to essentially self-supervise, and if effective, could be a helpful tool, as a DN would be able to learn on its own with no human intervention at all [2].

EXPERIMENTS

Our experimental DN setup followed that of our previous work. (1) The input is grabbed from a small, 135×135 pixel, mask on both the left and right images (green square in the figure below). (2) The hidden neurons are grouped into nine columns (inhibition zones), each column sharing the same initial receptive fields. (3) The input image is divided into 3×3 non-overlapping subwindows, of size 45 × 45 pixels, where each subwindow is the receptive field for the neurons in its respective column. (4) The number of hidden neurons is limited. We used our optimized DN version that uses the GPU and multiple CPU cores of the Android phone for faster update times; this allowed our training to approach real-time speeds.

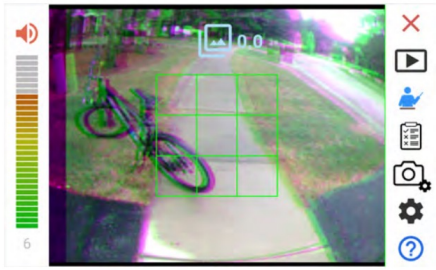

Fig 1. 3DEye software used for collecting and labeling data

The data in our experiment consisted of three stereo image sequences (Nav-1, Nav-2, Nav-3) collected in real-time from an outdoor walkway setting like the one in Fig. 1. Nav-1, Nav-2, and Nav-3 contained 6502, 6686 and 5508 frames, respectively. In this experiment, we used the even frames of each sequence for supervising, reinforcing, and practicing, and the odd frames were used as testing frames where we tested the DN's performance after training epochs in frozen mode, meaning that the DNs weights do not update with each update. The DNs multisensory ability allowed us to supervise the DN on 3 motors: a singular stereo-disparity, based on the closest object in the frame, a heading direction motor (left, straight, right), indicating what direction the user should turn to stay on the center of the sidewalk, and a stop/go motor, which tells the user whether they are about to walk off the path on their current course or not. These three motors were recorded using the 3DEye software, as seen in Fig 1, were labeled in real-time using the same software, and were verified afterwards.

We first directly supervised on the even frames of Nav-1. Here, the growth rate was configured so that half of the 750 neurons per column (750 × 9 = 6750 total hidden neurons) are activated in session 1, initialized sporadically throughout the first session. In session 3, all of the neurons are activated, and new neurons were initialized sporadically until all neurons were initialized by the end of the session. Our DN also had top-8 competition in each of the 3×3 = 9 columns, and a global top-3 competition in the motor area.

After our supervision sessions, we let the DN freely practice with no reinforcement over 3 epochs on Nav-3. This was to see how allowing the DN to freely practice improved performance on the test sequences.

In a separate experiment, after the same supervision sessions, we reinforced the DN over 3 epochs on Nav-3. In our reinforcement sessions, we punished the disparity motor if the error was above 4 and rewarded it if the error was less than 2. We then decreased these thresholds by one for each of the following reinforcement session. For our other two motors, we punished if they were incorrect and rewarded if they were correct. These motors had a top-1 competition in the motor area.

RESULTS

Session	Nav	Mode	Disparity	Heading	Stop/Go
1	1-even	Motor-S	0.00 px	0%	0%
2	1-odd	Frozen	1.27 px	10%	2%
3	2-even	Motor-S	0.00 px	0%	0%
4	2-odd	Frozen	1.43 px	15%	3%
5	1-odd	Frozen	2.19 px	17%	2%
6	3-odd	Frozen	2.74 px	20%	3%
7	3-even	Practice	2.71 px	21%	3%
8	3-odd	Frozen	2.70 px	23%	4%
9	3-even	Practice	2.67 px	24%	3%
10	3-odd	Frozen	2.70 px	23%	4%
11	3-even	Practice	2.68 px	26%	3%
12	2-odd	Frozen	1.65 px	15%	2%
13	1-odd	Frozen	2.31 px	21%	3%
14	3-odd	Frozen	2.67 px	26%	3%

Table 1. Average error through "lifetime" w/o reinforcement

Session	Nav	Mode	Disparity	Heading	Stop/Go
7	3-even	Reinforce	2.79 px	49%	39%
8	3-odd	Frozen	3.06 px	14%	4%
9	3-even	Reinforce	2.73 px	65%	53%
10	3-odd	Frozen	2.60 px	15%	4%
11	3-even	Reinforce	2.43 px	15%	3%
12	2-odd	Frozen	2.74 px	20%	3%
13	1-odd	Frozen	2.26 px	13%	4%
14	3-odd	Frozen	1.27 px	10%	2%

Table 2. Average error through lifetime w/ reinforcement

From our experimental data, we can see that a DN that underwent only practice sessions had slightly improved disparity errors after the practice sessions (see Table 2). This proves practice mode is effective and can be attributed to practice mode allowing neurons to update their weights to become more generalized, as seen in Fig 2.

From Table 2 we see that the reinforcement training on Nav-3-even improved the disjoint tests on Nav-3-odd. We also observe that such reinforcement training has negligible effects on the performance of other sequences (1 and 2), which means that their memory is not tangibly damaged. Ultimately, our multiple teaching modes is proven effective here, and the best performance was achieved with supervision and reinforcement, but practice mode also showed improvement.

Fig 3 shows a neuron that started as a noise neuron, representing a weak texture, and then re-learned a strong texture, representing a disparity of 4.

t	Age	Sensor Input	Disp	Bottom Up Weights
Nav1-4268	61		-3	
Nav1-4570	121		4	
Nav1-5595	181		4	
Nav2-3796	241		4	
Nav2-3895	301		4	

Fig 2. Updating weights over time in practice mode

t	Age	Sensor Input	Disp (Pred/True)	Bottom Up Weights
Nav1-187	1		~/-2	
Nav2-64	2		-4/-3	
Nav2-407	51		-4/-4	
Nav2-2320	101		-4/-3	

Fig 3. Noise neuron learning a concept over time

CONCLUSIONS

This work demonstrated that our system is capable of learning through multiple training modes and how our proposed training methods can improve training efficiency and performance of a DN. These methods have the potential to drive future autonomous systems.

REFERENCES

[1] J. Weng. Natural and Artificial Intelligence: Introduction to Computational Brain-Mind. BMI Press, Okemos, Michigan, second edition, 2019.

[2] J. A. Knoll, J. Honer, S. Church, and J. Weng. Optimal Developmental Learning for Multisensory and Multi-Teaching Modalities. In Proc. IEEE International Conference on Development and Learning, to be presented August, 2021.