The motor-action modulation role within the hierarchical predictive coding PredNet architecture

Junpei Zhong[1,3*] and Angelo Cangelosi[3] and Tetsuya Ogata[1,2]

***Abstract*— According to the sensorimotor contingencies, the predictive inference of our perception and the active motor action is encoded implicitly in the regularities between perception and action. We proposed a neural architecture of such regularities of active inference by a hierarchical neural architecture. In this architecture, this encoding emerges during the embodied learning process, when the action selector is driven to minimize the predictionerror in perception. Therefore, this predictive stream is modulated by the active motor actions and is continually updated by the bottom-up prediction error signals. In this way, the prediction forms the top-down streams which originally comes from the prior experience representing on the higher levels of this hierarchical architecture in an intransient way. This embodied model extends the PredNet Network with the motor action serving as a modulation role. This architecture is also examined on a robotic platform where the robot attempts to learn causal inference of predictive percept from actions while interacting with the objects. As a result, the *surprisal* can correct the sensorimotor behaviours, filling the gap of the predictive sensorimotor loop of the robot. We also analysed the robot's behaviours in a dynamically changing environment as well as the sensorimotor regularities in this hierarchical architecture.**

I. INTRODUCTION

Predictive process (PP) ([1, 2, 3, 4]) asserts that our sensorimotor loop works as a predictive machine, which works constantly as an active inference and keeps utilizing both active action and predictive perception to minimize the prediction error. Specifically, such error which the machine attempts to minimize is the difference between the posterior estimation and the truth, by changing its internal learning model ("perceptual inference" (see also [5] and [6]) or by action execution ("active inference", see also [7] and [8]). As such, perceiving the world (perceptual inference) and acting on it (active inference) are two processes that aim at minimizing the prediction error in the hierarchical architecture.

At the same time, this integrative process follows a bi-directional learning mechanism on each level of our hierarchical brain. Our mind needs to predict an incoming flow of sensory stimuli from the contextual factors in a top-down manner. Therefore, the neuronal representations on the higher level may generate predictions, and subsets of such prediction representations will be transmitted to the lower levels, to predict the upcoming neural activities on the lower level. In turn, this kind of predicting neural populations can be suppressed or inhibited by the prediction error which is transmitted in a bottom-up way. In this way, the internal world model in the brain has to be shaped by the statistical structure of the world which is perceived by the bottom-up flow. The world model infers the posterior of the next state or event following another based on the current or previous states. This hypothesis was firstly proposed by Helmholtz's unconscious inference ([9]), which claimed that the perception is cast as a process of unconscious inference, wherein perception is determined by both sensory inputs and our prior experience with the world.

The PredNet model ([10]) was considered to be the first model that utilised the PP concept into application, in which the camera video stream during driving can be predicted by the model. However, only the perception (video stream in this case) was considered in this PP framework. On the other hand, the execution of voluntary movements is also another factor while our mind is doing prediction. Within the synergistic relationship of perception and action, what we perceive (or think we perceive) is heavily determined by what we know and what we expect and execute, and what we know (or think we expect) is continuously modulated by our proprioception as well. Therefore, the world model within our PP framework results from the active execution of certain sensorimotor skills, rather than an internal representation merely from sensory signals.

A. Embodied Models

Given the multimodal aspect of the sensorimotor model, the construction of embodied models usually emphasize the embodied and the situated nature of the agents, to learn from interacting with the world ([11]). The predictive function of the internal model can range from short- and mid-term time-scale prediction/delay compensation to relatively long-term planning behaviors which emerge from the short-term simulations. The short-term predictive models are mostly related to sensorimotor control, especially consistency of visuomotor coordination (e.g. [12, 13]) or fast reaction (e.g. [14, 15]).

Some longer-term behaviors can emerge from such kind of short-term neural prediction as well. [16] and [17] studied how to apply internal model to control the actual motor actions. [18] also extended these models to learn imitation behaviours All of the three models built a forward predictive model to control the robot and acquire certain behaviours Similarly, a long-term planning behavour can also emerge

* Corresponding author: zhong@junpei.eu
[1]Artificial Intelligence Research Center, National Institute of Advanced Industry Science and Technology, Tokyo, Japan
[2]Lab for Intelligent Dynamics and Representation, Waseda University, Tokyo, Japan
[3]Centre for Robotics and Neural Systems, Plymouth University, Plymouth, PL4 8AA, United Kingdom

from internal simulation when the prediction is executed constantly ([19] and [20]). [21] reported experiments with a mobile robot implementing a two-level recurrent architecture to accomplish the linguistic and sensormotor task. An extension model has also been examined in a symbolic understanding tasks ([22]).

If we regard the unification of different time-scales of prediction, the Multiple Timescale Neural Network (MTRNN) proposed by [23] offers compressive model of such phenomena. The model is able to represent different temporal scales of sensorimotor information into the hierarchical structure of the the sensorimotor sequences, such as the language learning ([24, 25]) and object features/movements ([26]). As an extension of the MTRNN model with multiple modalities, the multiple spatio-temporal scales RNN (MSTRNN) ([27]) integrates the MTRNN and convolutional neural networks ([28, 29]). It includes two modalities: both the temporal properties as well as the spatial receptive field sizes in different levels. The PredNet ([10]) also holds a similar concept of using the convolutional network to capture the local features of the visual streams, but the temporal constraints are implicitly hidden. Moreover, both models use only the information from visual stream for recognition/prediction but do not incorporate any action-guided predictions. This is the main motivation we are proposing for a new action modulated predictive model.

II. Model

Compared with the PredNet, the AFA-PredNet (Action Feedback Augmented Predictive Network) architecture (Fig. 1) proposed in this paper further integrates the motor action as an additional signal which modulates the top-down generative process via an attention mechanism. This modulation role is similar to the integration process, with perception prediction while having the active motor action as a consideration.

Similar to the hierarchical architecture in the sensorimotor integration and the deep learning architecture, the AFA-PredNet network consists of a series of repeated stacked modules in a hierarchical way, which attempt to make local predictions of the visual inputs. In general, the AFA-PredNet is functionally organized as an integration with two networks: the left part is equivalent to a generative recurrent network, while the right part is a standard convolutional network. Each layer of the network consists of four basic parts: a generative unit (*GU*, green) containing the recurrent convolutional networks with the motor modulated unit (*MM*, grey), a discriminative unit (*DU*, blue) containing convolutional networks (CNN) and the error representation layer (*ER*, red). The generative unit, *GU*, is usually a recurrent network that generates a prediction of the next time-step from the current input. Here, the convolutional LSTM ([30, 31]) is employed to generate the local prediction in the image region. We employ a number of independent recurrent units on one layer of the *GU* unit. During training with various action-perception paring occasions, each of these units implicitly memorizes different possibilities of the prediction (e.g. the moving direction) with respect to the motor action in a self-organized way.

The *DU* network discriminates the errors by calculating the difference between convolutional output of the predicted signal from *GU* as well as the bottom-up signal as an error representation, *EL*, which is split into separate rectified positive and negative error populations. The error, *EL*, is then passed forward through a convolutional layer to become the input to the next layer.

A. Neural Dynamics

In the following section, we denote the indices of these perception input image as i_t, and the target of the network prediction at the lowest level is set to the actual percept at the next time-step i_{t+1}. We directly put the image as the input of the lowest layer, layer 0, so the input of the layer 0, X_0, equals to the actual image data $X_0^t = i_t$.

The targets for higher layers at time-step t is denoted as $X_l(t)$. Except layer 0, $X_l(t)t$ is obtained by the higher level representation of the deep convolutional layer, which follows a usual calculation process of the convolutional network as shown in Eq. 1: the convolution kernel, the rectified linear unit (ReLU) calculation and the max-pooling are sequentially used. This bottom-up process using convolutional network to extract the local features of the error.

At the *GU* unit, the generative process is determined by the representation from the recurrent connection (i.e. from the previous time-step) X, the bottom-up error $E_l(t-1$ as well as the top-down prediction $R_{l+1}(t)$. Such a prediction in a convolutional LSTM is calculated as Eq. 4: a deconvolution is used to reconstruct a larger size of the (predicted) representation $\hat{A}$ after an Rectifield unit calculation (ReLU) (Eq. 2).

To avoid the drawback of the ReLU which only capture only the positive and negative error, the error representation $E_l(t)$ is calculated from the positive and negative errors (Eq. 3), as the original PredNet does. The modulation role of motor actions are represented as an multiple layer perceptron (MLP) here, whose output explicitly represents as the movement factors of multiple recurrent units (*GU*) of the higher level (Eq. 5), which are further multiplied by all the possible recurrent *GU* units. Such motor modulated prediction can be further augmented by the attention model, such that the predicted perception can added in order to build a closed loop in the sensorimotor integration, where the context of the perception can be considered to select a proper motor action.

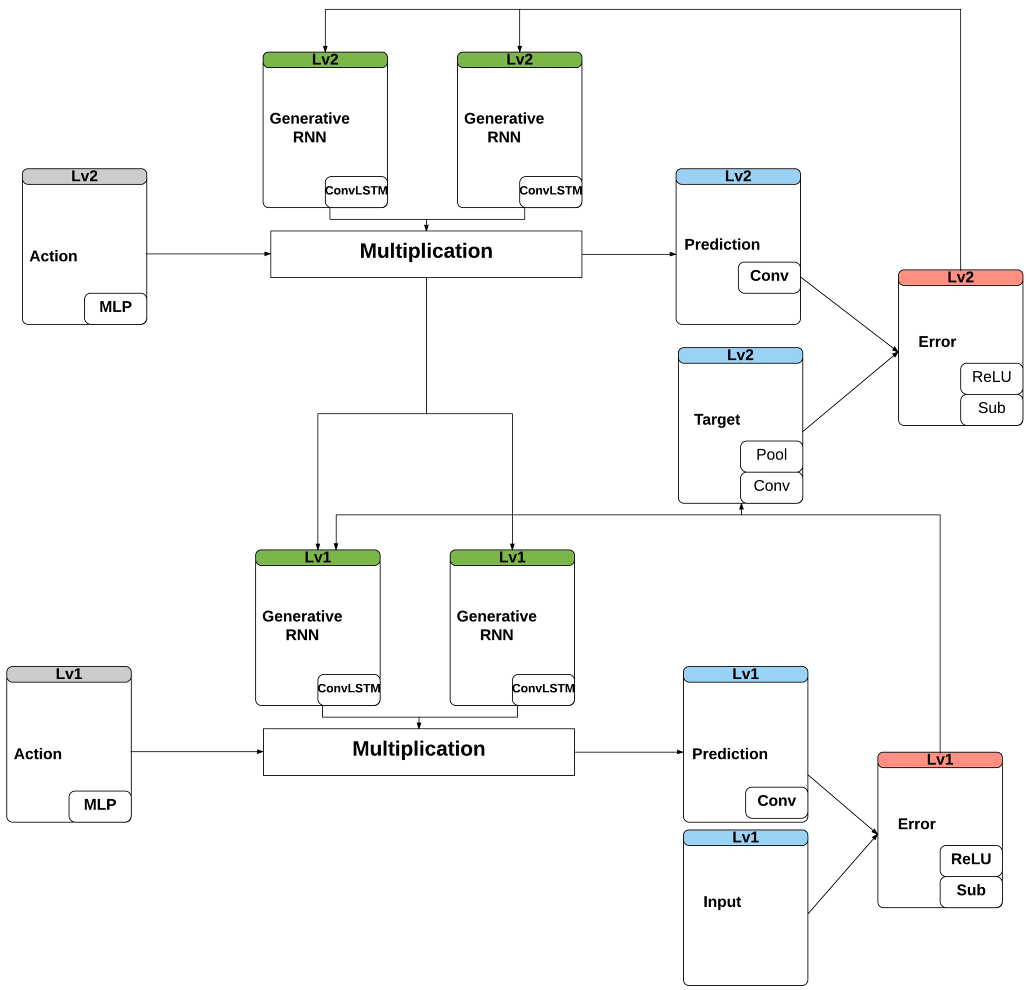


Fig. 1: A 2-layer AFA-PredNet

$$X_l(t) = \begin{cases} i(t), & \text{if } l = 0, \\ MAXPOOL(f(Conv(E_{l-1}(t)))), & l > 0 \end{cases} \tag{1}$$

$$\hat{X}_l(t) = f(Conv(R_l(t))) \tag{2}$$

$$E_l(t) = [f(X_l(t) - \hat{X}_l(t)); f(\hat{X}_l(t) - X_l(t))] \tag{3}$$

$$R_l^d(t) = ConvLSTM(E_l(t-1), R_l(t-1), DevConv(R_{l+1}(t))) \tag{4}$$

$$R_l(t) = MLP(a(t)) \times R_l^d(t) \tag{5}$$

$$\tag{6}$$

where $f(\cdot)$ is an activation function of the neurons, which we apply ReLu function to ensure a faster learning in back-propagation, $X(\cdot)_l^t$ is the neural representation of the level l at time t. The representation on the *EL* layer l is $E(\cdot)_l$. The $MAXPOOL$, $Conv$, $ConvLSTM$ and MLP are the corresponding neural algorithms.

The overall algorithm for learning a whole sequence as showed in Algorithm 1:

III. Experiment Results

A. Minimalistic World

Our first experiment was started by using a set of artificially generated visual input data which mimics an moving object perceived from our visual system, i.e. its position changes quickly at every timestep. In such scenario, the external movements of an object are manipulated by the voluntary active motor action, e.g. the robot moves an object toward left or right. So the motor commands cause the

Data: $i(t)\&a(t) \in data$
while $error > threshold$ or $iteration > maximum\_iteration$ **do**
    for $t \leftarrow 0$ **to** T **do**
        for $l \leftarrow 0$ **to** L **do**
            if $l == L$ **then**
                $R_l^d(t) = ConvLSTM(E_l(t-1), R_l(t-1);$
                $R_l^d(t) = ConvLSTM(E_l(t-1), R_l(t-1), DevConv(R_{l+1}(t)));$
            else
                $R_l(t) = MLP(a(t)) \times R_l^d(t);$
            end
        end
        /* Generative (top-down) Process */
        for $l \leftarrow L$ **to** 0 **do**
            $\hat{X}_l(t) = f(Conv(R_l(t)));$ $E_l(t) = [f(X_l(t) - \hat{X}_l(t)); f(\hat{X}_l(t) - X_l(t));$
                /* Discriminative (bottom-up) Process */
        end
    end
end

Algorithm 1: AFA-PredNet Computation

changes in the visual perception in this case. This minimalistic set up sketches a tracking scenario which is usually perceived from the visual receptors.

In this dataset, the size of the input space of the visual field is 8×12 and only one object appears at one unique position in any time-step. The training data set comprises two directional movements (horizontally or vertically) covering all of the possible sequences of all objects. The direction of the movement, either toward right or down, is determined by an action vector containing two neurons. For instance, the Fig. 2 and Fig. **??** contain an activation moving toward right and toward bottom.

A 2-layer AFA-PredNet was utilised for training. In the training process, the target data is the one-step ahead prediction of the input data. In the experiment, the maximum iteration is set to be $100,000$, learning rate is set to be $\eta = 0.001$, the number of hidden neurons in the action MLP is set to 4. Other parameters of the CNN are fixed and shown in Tab. I.

Parameters	Value
Kernel	3×3
Padding	1
Pooling	2×2

TABLE I: CNN parameters

After training, we manually set the action vector to be $[1, 0]$, which indicates that the motor action is from left toward right, and $[0, 1]$, which indicating that the object movement from up toward down) in order to examine the network performance. While we assume the object movement is from left toward right, the original images we selected from the central location $(4, 6)$ are shown below (Fig. 2 and Fig. 4):

To examine another movement direction, we set the motor action vector to be $[0, 1]$. We also pick up a series of original images from location $(4, 6)$ as shown in Fig. 4. The predicted images are shown in Fig. 5.

To further investigate the functions of the multiple *GU* units, we also illustrate the neural outputs of the multiple recurrent units *GU* on each layer. The reason for doing so is to see what do their neural activities represent in the embodied context, i.e. given the action vector a and a sequence of images i. Furthermore, from those representations, we can also infer the functions of the MLP network. In order to do this, we feed the network with a sequence of the pre-trained images and set the action unit to be $[0, 1]$. Then we visualised the neural representation of the *GU* from $(5, 4)$. As shown in Fig. 6, although there are some noised representation, the two *GU* units represent two possibilities of the next movement, around the previous activated position $(5, 4)$. After that, the MLP network probably play a role as an action selector and filter to predict the next movement, given the action vector input.

B. Line Tracer Robot

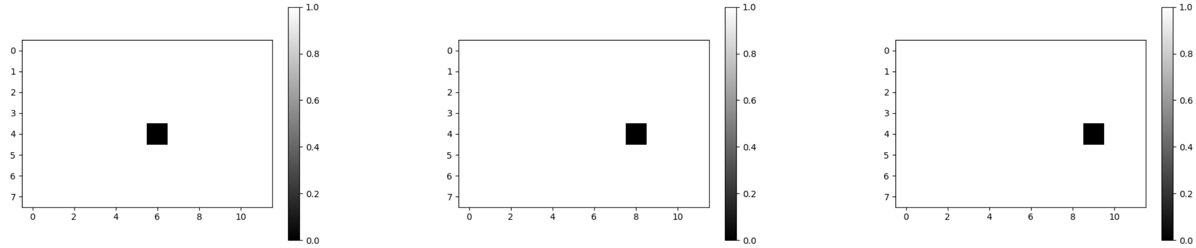
To examine the network performance in a robotic system, we recorded the simulation data about the line tracer robot car from the VRep simulator ([32]). In this scenario, the robot car equips three vision sensors as well as three Line Finder sensors. With these sensors, the robot is able to adjust the velocities of its wheels to follow the line. Using VRep as a tool, what we do in this experiment ware

- collect wheel velocity data and camera data;
- use this data to train and verify the network offline.

Therefore, with the proposed AFA-PredNet, we are able to predict the images which will appear in the vision sensor according to the velocity output of the two wheels at the next time-step. To gather the data, we capture the grey-scale images with size of 8×12 pixels from the vision sensor in the middle every 0.02s. Fig. 8 showed 5 sample images in every 0.8 s, which the white shades are the line followed by the robot. Inputs of the action vector a are the velocities of the robot car.

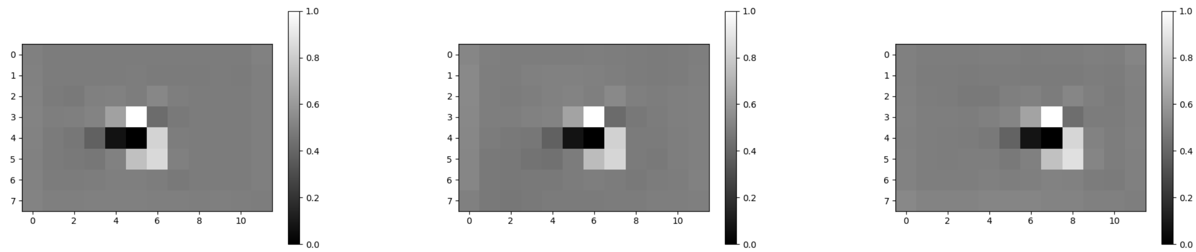
Training of the AFA-Prednet for the line tracer robot followed a similar procedure as the previous experiment. The target data is one time-step ahead of the input image (i.e. the next image in around 0.02 second). We used a 3-layer AFA-PredNet, with 4 hidden units in the MLP network.

After the training, we feed the sequence of the observed images (Fig. 8) to the input and the sequence of the wheel velocities to motor action units. The Fig. 9 illustrates the predicted images corresponding to the original inputs, in which we can observe that the AFA-PredNet can generate a distinguishable one time-step prediction for the vision system of the Line Tracer robot.



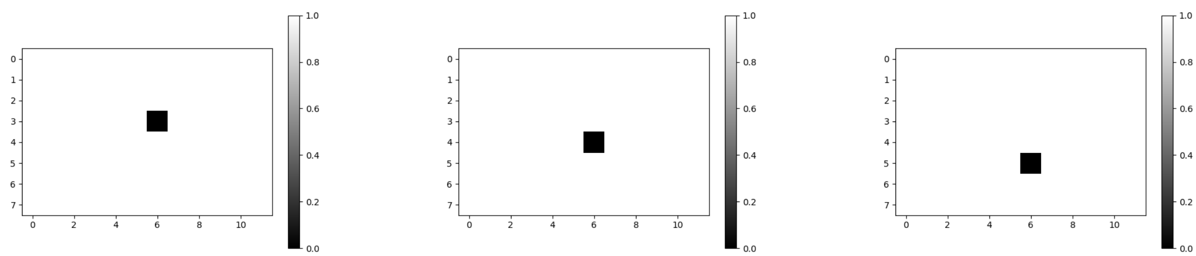
(a) (b) (c)

Fig. 2: Original Images (movement from left to right)



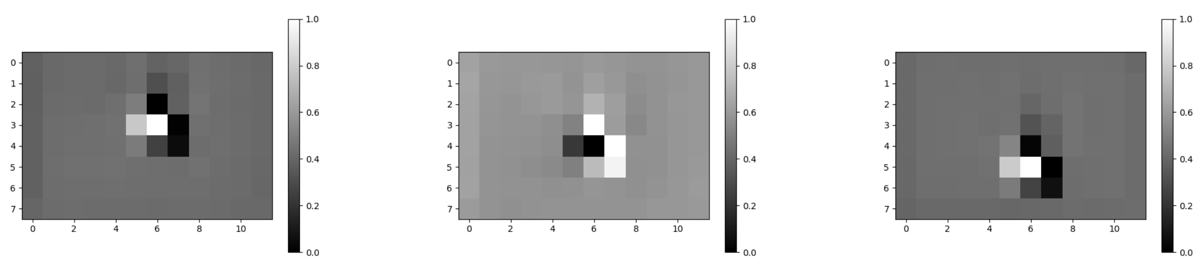
(a) (b) (c)

Fig. 3: Predicted Images (movement from left to right)



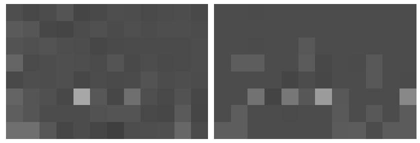
(a) (b) (c)

Fig. 4: Original Images (movement from top to bottom)



(a) (b) (c)

Fig. 5: Predicted Images (movement from top to bottom)



(a) (b)

Fig. 6: Representation of Generative Units (Layer 0)

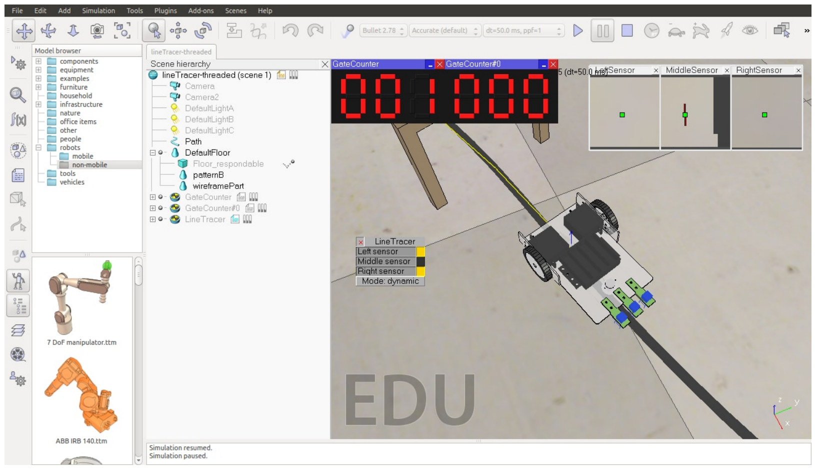
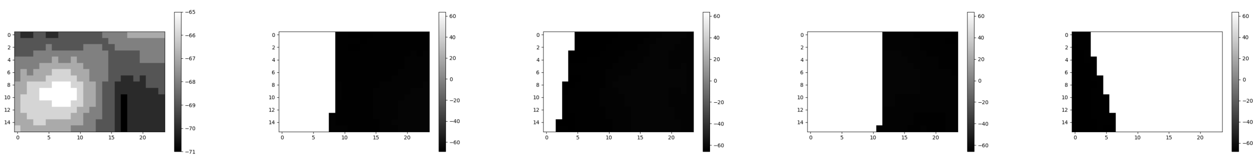
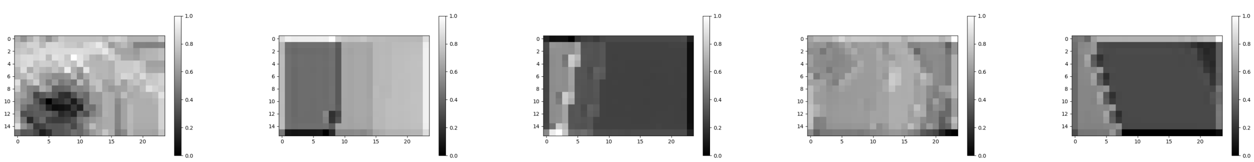


Fig. 7: Data Collected from VRep Simulation



(a) 0 second (b) 0.8 seconds (c) 1.6 seconds (d) 2.4 seconds (e) 2.8 seconds

Fig. 8: Image Samples from the Middle Vision Sensor



(a) 0 second (b) 0.8 seconds (c) 1.6 seconds (d) 2.4 seconds (e) 2.8 seconds

Fig. 9: Predicted Images from the Same Sequence

IV. DISCUSSIONS AND SUMMARY

A. From Predictive Perception to Planning

The feedback affecting sensory input can be regarded as a kind of predictive information retrieved from the internal memory ([33]). Based on PC, in the hierarchical architecture, the feedback signals (especially the top-down signals) predict the forthcoming sensory input, while the sensory-driven bottom-up signals only deliver the error of the estimation. As we concluded, these functions are not independent; instead they are processes that happen at the same time and integrate with each other. They are performed with the similar Bayesian inference and are always interchanging prior knowledge on the cognitive processes level.

Similar, on the cognitive process level, if such kind of prediction lasts as a closed-loop, long enough in a hierarchical way, plays as a mental simulation about future events. Such a prediction is also about multi-modality too, which captures the structural regularities in the modality, spatial and temporal space ([34]), to accomplish the tasks about decision making and planning. As such, the difference between the sensorimotor prediction and the planning behaviour is just a matter of time-scale.

It should be admitted that it is a trivial explanation regarding their difference. As specified at [35], such a planning process inherited from the predictive process only exists, when:

1) the specific goal is already determined at the very first beginning;
2) at a short- or mid-term planning problem.

For more complex planning problems, such as the multi-objective optimization problem (e.g. Traveller Salesman Problem, TSP), it needs a higher level of cognitive computation power and time. Nevertheless, from the engineering perspective, the short- and mid-term planning is sufficient in some short- and mid-term planning applications, e.g. autonomous driving, where the PredNet model was already examined to predict the next frame of the vehicle camera.

To sum up, the top-down prediction may happen through the whole the brain from the cogntive function to the sensorimotor processes is essential as they have the following benefit on the lower-level peripheral perception functions:

1) The target of the feedback pathways in perception is applied for sensory prediction. It is realized by extracting cues from multimodal or amodal perception via feature extraction (e.g. by the early visual system) which becomes a prior. Then, the posterior estimation is applied to the next predictive perception.
2) If there is a difference between the posterior estimation and the current receptor signals, the percept may be derived as a combination of the two to avoid the fluctuation caused by neuronal or receptor noise. On the other hand, the error signals are also transmitted from bottom-up signals to further act as a prior in the perception cues.

B. Summary

We proposed an embodied model to address the hierarchical predictive coding (PC) architecture. Integrating both perception and action, the model is able to predict the percept according to the motor action executed. This hierarchical model is based on four basic modules:

1) the generative unit (*GU*) which contains convolutional LSTM to generate prediction in a top-down manner;
2) the motor modulated unit (*MM*) which uses MLP to convert the motor information to object movement information;
3) the discriminative unit (*DU*) employing convolutional networks (CNN) to discriminate the bottom-up information and to give feedback to *GU*.
4) the error representation layer (*EL*) to calculate the difference between prediction and real inputs

Although only the preliminary experiments were conducted, a short-term prediction of perception has been testified in a minimalistic world model as well as a line tracing robot simulation. We also examined some intriguing representation in the *GU* units.

At the next step, extensions can be made in the consideration of doing prediction based on the context of percepts as well as building prediction at longer time-scale.

APPENDIX

A Pytorch implementation of AFA-PredNet can be found on Github[1]

ACKNOWLEDGMENT

The research was supported by Japan New Energy and Industrial Technology Development Organization (NEDO).

REFERENCES

[1] A. Clark. "Whatever next? Predictive brains, situated agents, and the future of cognitive science". In: *Behavioral Brain Sciences* (2012), pp. 1–86.

[2] R. P. Rao and D. H. Ballard. "Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects". In: *Nature neuroscience* 2.1 (1999), pp. 79–87.

[3] K. Friston. "Learning and inference in the brain". In: *Neural Networks* 16.9 (2003), pp. 1325–1352.

[4] K. Friston. "A theory of cortical responses". In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 360.1456 (2005), pp. 815–836.

[5] E. M. Segal and T. G. Halwes. "The influence of frequency of exposure on the learning of a phrase structural grammar". In: *Psychonomic Science* 4.1 (1966), pp. 157–158.

[6] K. Friston and S. Kiebel. "Cortical circuits for perceptual inference". In: *Neural Networks* 22.8 (2009), pp. 1093–1104.

[7] K. Friston, J. Mattout, and J. Kilner. "Action understanding and active inference". In: *Biological cybernetics* 104.1 (2011), pp. 137–160.

[8] G. Pezzulo, F. Rigoli, and K. Friston. "Active Inference, homeostatic regulation and adaptive behavioural control". In: *Progress in Neurobiology* 134 (2015), pp. 17–35.

[9] H. Von Helmholtz. "Concerning the perceptions in general". In: *Treatise on physiological optics,* (1866).

[10] W. Lotter, G. Kreiman, and D. Cox. "Deep predictive coding networks for video prediction and unsupervised learning". In: *arXiv preprint arXiv:1605.08104* (2016).

[1] https://github.com/jonizhong/afa_prednet.git

[11] R. A. Brooks. “How to build complete creatures rather than isolated cognitive simulators”. In: *Architectures for intelligence* (1991), pp. 225–239.

[12] E. von Holst and H. Mittelstaedt. “The reafference principle: Interaction between the central nervous system and the peripheral organs. Selected Papers of Erich von Holst: The Behavioural Physiology of Animals and Man”. In: (1950).

[13] R. C. Miall and D. M. Wolpert. “Forward models for physiological motor control”. In: *Neural networks* 9.8 (1996), pp. 1265–1279.

[14] N. L. Cerminara, R. Apps, and D. E. Marple-Horvat. “An internal model of a moving visual target in the lateral cerebellum”. In: *The Journal of physiology* 587.2 (2009), pp. 429–442.

[15] J. Zhong, C. Weber, and S. Wermter. “A Predictive Network Architecture for a Robust and Smooth Robot Docking Behavior”. In: *Paladyn. Journal of Behavioral Robotics* 3.4 (2012), pp. 172 –180.

[16] D. M. Wolpert, Z. Ghahramani, and M. I. Jordan. “An internal model for sensorimotor integration”. In: *Science* (1995), pp. 1880–1880.

[17] D. M. Wolpert and M. Kawato. “Multiple paired forward and inverse models for motor control”. In: *Neural Networks* 11.7-8 (1998), pp. 1317–1329.

[18] Y. Demiris and B. Khadhouri. “Hierarchical attentive multiple models for execution and recognition of actions”. In: *Robotics and autonomous systems* 54.5 (2006), pp. 361–369.

[19] H. Hoffmann. “Perception through visuomotor anticipation in a mobile robot”. In: *Neural Networks* 20.1 (2007), pp. 22–33.

[20] R. Möller and W. Schenck. “Bootstrapping cognition from behaviora computerized thought experiment”. In: *Cognitive Science* 32.3 (2008), pp. 504–542.

[21] Y. Sugita and J. Tani. “Learning semantic combinatoriality from the interaction between linguistic and behavioral processes”. In: *Adaptive Behavior* 13.1 (2005), p. 33. ISSN: 1059-7123.

[22] J. Zhong, A. Cangelosi, and S. Wermter. “Towards a self-organizing pre-symbolic neural model representing sensorimotor primitives”. In: *Frontiers in Behavioral Neuroscience* 8 (2014), p. 22.

[23] Y. Yamashita and J. Tani. “Emergence of functional hierarchy in a multiple timescale neural network model: a humanoid robot experiment”. In: *PLoS Computational Biology* 4.11 (2008), e1000220.

[24] T. Ogata and H. G. Okuno. “Integration of behaviors and languages with a hierarchal structure self-organized in a neuro-dynamical model”. In: *Robotic Intelligence In Informationally Structured Space (RiiSS), 2013 IEEE Workshop on*. IEEE. 2013, pp. 89–95.

[25] J. Zhong, A. Cangelosi, and T. Ogata. “Toward Abstraction from Multi-modal Data: Empirical Studies on Multiple Time-scale Recurrent Models”. In: *arXiv preprint arXiv:1702.05441* (2017).

[26] J. Zhong et al. “Sensorimotor Input as a Language Generalisation Tool: A Neurorobotics Model for Generation and Generalisation of Noun-Verb Combinations with Sensorimotor Inputs”. In: *arXiv preprint arXiv:1605.03261* (2016).

[27] H. Lee, M. Jung, and J. Tani. “Recognition of visually perceived compositional human actions by multiple spatio-temporal scales recurrent neural networks”. In: *arXiv preprint arXiv:1602.01921* (2016).

[28] Y. LeCun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[29] J. Donahue et al. “Long-term recurrent convolutional networks for visual recognition and description”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 2625–2634.

[30] S. Hochreiter and J. Schmidhuber. “Long short-term memory”. In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[31] X. Shi et al. “Convolutional LSTM network: A machine learning approach for precipitation nowcasting”. In: *Advances in neural information processing systems*. 2015, pp. 802–810.

[32] E. Rohmer, S. P. Singh, and M. Freese. “V-REP: A versatile and scalable robot simulation framework”. In: *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. IEEE. 2013, pp. 1321–1326.

[33] J. R. Anderson and L. J. Schooler. “The adaptive nature of memory.” In: (2000).

[34] M. Toussaint. “Probabilistic inference as a model of planned behavior.” In: *KI* 23.3 (2009), pp. 23–29.

[35] D. Basso. “Planning, prospective memory, and decision-making: three challenges for hierarchical predictive processing models”. In: *Frontiers in psychology* 3 (2013), p. 623.