OPEN ACCESS
Journal of Artificial Intelligence and Consciousness
Vol. 9, No. 1 (2022) 29−57

DOI: 10.1142/S2705078521500211

World Scientific
www.worldscientific.com

A Design of Global Workspace Model with Attention: Simulations of Attentional Blink and Lag-1 Sparing

Wenjie Huang[*,‡], Antonio Chella[†,§] and Angelo Cangelosi[*,¶]

[*]*Department of Computer Science, University of Manchester*
Manchester, M13 9PL, UK

[†]*Department of Engineering, University of Palermo*
Palermo 90128, Italy
[‡]*wenjie.huang@postgrad.manchester.ac.uk*
[§]*antonio.chella@unipa.it*
[¶]*angelo.cangelosi@manchester.ac.uk*

Received 21 July 2021
Accepted 3 September 2021
Published 13 October 2021

There are many developed theories and implemented artificial systems in the area of machine consciousness, while none has achieved that. For a possible approach, we are interested in implementing a system by integrating different theories. Along this way, this paper proposes a model based on the global workspace theory and attention mechanism, and providing a fundamental framework for our future work. To examine this model, two experiments are conducted. The first one demonstrates the agent's ability to shift attention over multiple stimuli, which accounts for the dynamics of conscious content. Another experiment of simulations of attentional blink and lag-1 sparing, which are two well-studied effects in psychology and neuroscience of attention and consciousness, aims to justify the agent's compatibility with human brains. In summary, the main contributions of this paper are (1) Adaptation of the global workspace framework by separated workspace nodes, reducing unnecessary computation but retaining the potential of global availability; (2) Embedding attention mechanism into the global workspace framework as the competition mechanism for the consciousness access; (3) Proposing a synchronization mechanism in the global workspace for supporting lag-1 sparing effect, retaining the attentional blink effect.

Keywords: Consciousness; Global Workspace; Attention; Attentional Blink; Lag-1 Sparing; Synchronization.

1. Introduction

In the past decade, artificial intelligence (AI) research has rapidly advanced with the help of neural networks or deep learning. Though the domain-specific (narrow) AI is

the nowadays mainstream of the AI community, people are witnessing the resurrection of artificial general intelligence (AGI) [Chella and Manzotti, 2007, 2011, 2013]. This branch is referred to as the modeling of human-level intelligence, which is generalizable to various fundamental areas of AI [Goertzel, 2007]. Within this area, machine consciousness is one of the key topics, involving efforts from multiple disciplines like neuroscience, philosophy, psychology, cognition and computer science. Specifically, for researchers centered in computer science like the authors, the work in this area tend to develop computational models to duplicate neuroscientific mechanisms or human behaviors related to consciousness [Chella and Manzotti, 2013; Reggia, 2013].

There are a variety of theories for such development purposes [Reggia, 2013; Chella *et al.*, 2019], but so far nobody has successfully demonstrated consciousness with artificial systems. Regarding this research dilemma, three causes are identified here for why machine consciousness is not yet developed or accounted for in artificial systems. The first one is concerning the definition of consciousness. Generally, consciousness still resists a widely accepted definition [Vimal, 2009; Reggia, 2013; Chella *et al.*, 2019]. Such absence of consensus implies elusive research goals and requirements in this area [see, for attempt on this, Manzotti and Chella, 2020], and divergent assessments for machine consciousness. This problem directly results in the failure in constructing the complete set of specifications of consciousness for the development and leads to another trap some researchers are in. This trap, we call the *isolation trap*, is focusing effort strictly on individual modules or mechanisms to account for the emergence of consciousness [Holland and Goodman, 2003; McDermott, 2007; Tsuchiya and Adolphs, 2007; Reggia *et al.*, 2020; Blum and Blum, 2021]. Being stuck in this pitfall, people are prone to neglect the interactions between various processes in the human or other animal brains for the emergence of consciousness. Thirdly, the vague boundary between intelligence and consciousness remains confusing though it is not yet explicitly identified in the literature. It appeals for either conceptual discrimination to avoid tackling the big problem of intelligence at once or directly absorbing consciousness into intelligence to eliminate the theoretical redundancy.

Any research targeting one or more of the three problems would contribute to the progress in the area of (artificial) consciousness. This work exactly focuses on seeking a way out of the *isolation trap*. For this purpose, we argue that consciousness is a property arising from the intelligent system as a whole, instead of certain modules or mechanisms. A feasible approach to achieve this is incrementally developing modules or processes toward a conscious system.

For the incremental development, this paper attempts to integrate the global workspace theory (GWT) and the attention mechanism as the start point. For GWT, it has been acknowledged as a general framework for consciousness to happen, playing as a basis for many models and studies [Franklin, 2003; Shanahan, 2006; Baars and Franklin, 2009]. This allows to increasingly incorporate modules and processes into this architecture, compatible with the incremental scheme. By recruiting this modeling framework, the agent can attend to only one event out of the

overwhelming information at a certain time, and broadcast it to all the parallel specialized processors. This is attributed to the three features postulated in GWT, namely (i) specialized processors, (ii) competition for entering consciousness and (iii) global availability [Baars, 1993; Shanahan, 2006]. Such features are only general guidance, receptive to adaptations. One of the variant structures is proposed by Shanahan and Connor [2008] called global neuronal workspace, emphasizing the competitive access and broadcast dynamics of the brain based on GWT. The novelty of this work is that the global workspace is broken into individual nodes for each module, which efficiently connects widely distributed regions in the brain together. In other words, it implements GWT into working spaces located in distributed regions rather than the originally proposed "theatre stage" [Baars, 1993] which is independently situated in a certain region of the brain. Despite the topological distinction, the role and functions of the global workspace nodes are not changed.

Inspired by this attempt of Shanahan and Connor, this work makes a radical adaptation of GWT. This attempt focuses on the feature of global availability in GWT, which was verified by the discovery that conscious perceptions trigger widespread cortical activity [Dehaene *et al.*, 2001]. The global availability feature indicates, taking the model of Shanahan and Connor [2008] as an example, all the specialized processors are responsive to the broadcast information and trigger the next competition for access to consciousness. However, we argue that not all modules need to be responsive to the attended information, by which much the redundant computation could be reduced. One possible implementation for this purpose is that the information represented in the global workspace is transmitted to a subset of all the specialized modules, while other modules remain irresponsive. To achieve this purpose, we propose separated global workspace nodes (Fig. 1).

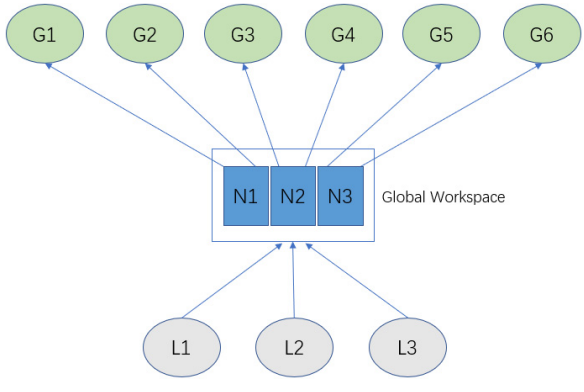

Fig. 1. (Color online) Global workspace framework with separated workspace nodes. The gray circles are local modules for receiving stimuli and modality-specific processing (e.g., speaking perception and word processing; image perception and recognition). The global workspace consists of separated nodes (blue rectangles). The forwarded signals from local modules compete to exclusively occupy the global workspace. Then, the signals represented in the global workspace nodes are, respectively, only available to a subset of the global modules (green circles).

In this framework, though we agree access to the global workspace is the fundament for consciousness to happen, we avoid terms like conscious processors or modules. This is because consciousness, in our opinion, arises from the whole system rather than certain modules. The distinction of the global modules from the local ones is that multiple signals may require the responses from them simultaneously, namely, they are shared.

Another issue that remained unspecified is the competitive access to consciousness within this architecture. For this, the attention mechanism is proposed as an access gate to consciousness to determine which stimulus could be responded to [Baars, 2002]. Attention is usually treated as a key correlate to consciousness. Its role in achieving consciousness has been studied by many scientists [Haikonen, 2003; Tinsley, 2008; Starzyk and Prasad, 2011]. In this work, we take the stand that attention is not equating consciousness but a premise allowing consciousness to happen. To integrate these two theories, we attempt to embed the attention mechanism into the GWT. This is because the competitive mechanism of attention provides a feasible approach to the GWT-based architecture on distinguishing conscious/global and subconscious/local activities [Baars, 1993]. A detailed view of the attention mechanism is provided by Knudsen [2007], which incorporates bottom-up filtering, top-down attention, competitive selection and working memory. The bottom-up information stream is driven by the features of the stimuli perceived. After that, a voluntary (top-down) sensitivity control is imposed on specific fragments of the bottom-up flow to increase or inhibit their competitiveness for entering the working space. This top-down signal is generated by the working space for tasks demands [Buschman and Miller, 2007; Taylor, 2007]. To adapt this scheme into the global workspace framework, the bottom-up filtering is embedded in the local modules, and the working memory is mapped to the global workspace and global modules in Fig. 1. Figure 2 is a simplified structure for illustrating both top-down and bottom-up attention.

With these two theories, GWT and attention mechanism combined, we implement a novel GWT-based model, exploiting the attention mechanism as the competitive access (formally introduced in Sec. 2). This model consists of modules of vision, audio, hand-motor, eye-motor, bottom-up and top-down attention and global workspace. Compared to those existing work, this model is designed with separated global workspace nodes to reduce unnecessary computation, and the synchronization mechanism to improve the efficiency of information processing. On the other hand, such implementation retains the three features of the GWT. As illustrated in Fig. 1, when all the nodes of the global workspace are active with valid signals, it functions exactly like the original idea, broadcasting the information to all the global modules. Otherwise, because the global processes only respond to the information from certain global workspace nodes, it avoids unnecessarily forwarding information to irresponsive modules and reduces redundant computation.

To validate this proposed model, we design two experiments. One is to simulate the agent's ability to attend to specific stimuli, dynamically change its attentional content and exploit the perceived information (e.g., generation of top-down attention

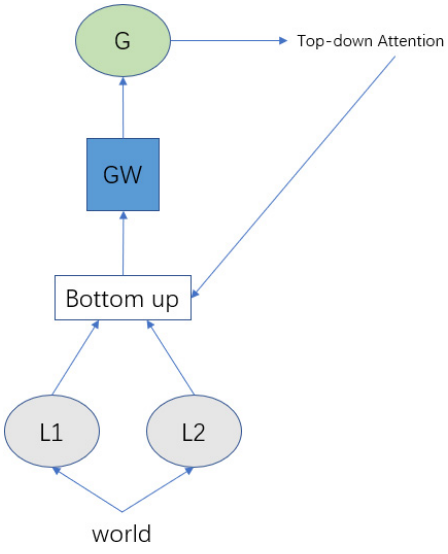

Fig. 2. Attention scheme adapted in GWT-based framework. The agent perceives stimuli via local modules, where the saliences of the stimuli are purely computed based on the signal features (e.g., pixel intensity, voice magnitude). The top-down signal represents the agent's voluntary interference of attention to certain stimuli depending on tasks (e.g., enhance the salience of auditory stimuli while talking).

interference, motor reaction). Another focuses on the dynamics of the information processing within the system, which is inspired by the work of Shanahan [2006, 2012], and Dehaene *et al.* [2003]. In this second experiment, attentional blink and lag-1 sparing effects are simulated since they are closely related to attention and consciousness. Attentional blink happens when the agent is exposed to a rapid stream of stimuli [Raymond *et al.*, 1992]. Specifically, when a second stimulus is presented between about 100 and 500 ms after the first stimulus, then the agent is unable to consciously perceive the second one as if she is temporarily blinking. This effect was well-simulated by Dehaene *et al.* [2003] within a global workspace framework and attributed to the competitive access to the workspace. However, they did not implement the competition process explicitly. For called lag-1 sparing effect, it implies that if the second stimulus immediately happens after the first one, they compete for attention as they are simultaneously happening and the second one tends to win if they are of equal strength [Hommel and Akyürek, 2005]. To our best knowledge, there is no existing work focusing on this, leaving it unclear that if this effect could be successfully demonstrated in the architecture we discussed here.

By these experiments, this work aims to (1) study the role of attention in achieving consciousness; (2) prove that the model implemented in this work is compatible with the conscious brains of humans; (3) identify the future direction for the incremental development toward a conscious machine.

In the following sections, Sec. 2 gives brief introductions of the model, the dataset and the training of networks, and the tasks configurations; Sec. 3 demonstrates the

simulation experiments and the results; the work is concluded in the last section with: discussion of the experiment results (Objectives 1, 2), the comparison to the existing work helping identify the advantages of this work, the limitations of our work and future work directions (Objective 3).

2. Methodology

In this section, we firstly display the design of the model based on the proposed novel GWT framework and attention mechanism. Then, we train the networks and present the data. After this, the two experiment tasks are introduced.

2.1. *Model*

The structure of the agent is organized in two levels: the local one for modality-specific processing and the global one which represents the computational resource under competition. The architecture is shown in Fig. 3.

While running, this agent is exposed to both visual (image) and auditory (speaking) stimuli in the environment. When the agent receives stimuli from either or

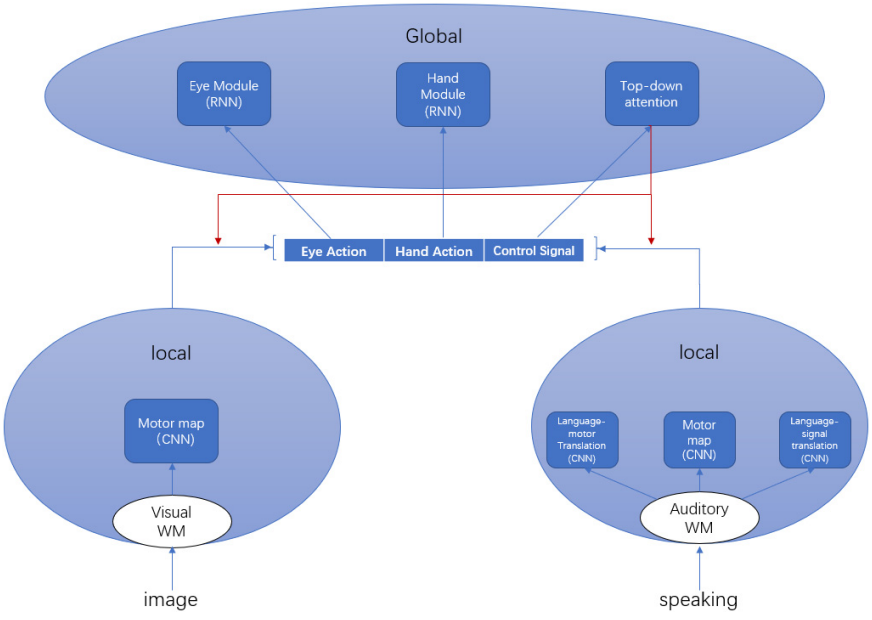

Fig. 3. (Color online) The structure of the implemented model. Each local module consists of working memory and modality-specific processes. This design is partly inspired by the work of Dehaene *et al.* [2001]. According to their repetitive suppression experiment, the words failing to enter consciousness still activate the word-specific processing area. Based on this, we hypothesize that all the modality-specific activities happen locally. The blue arrows indicate the directions of information flow, which are in a bottom-up manner. Conversely, the red arrows indicate the top-down attention, imposed on the signals forwarded by local modules. In addition, the three nodes [eye action; hand action; control signal] comprise the global workspace. The details are explained in the appendix.

both of the two modalities, the stimuli are firstly pre-processed by the modality-specific working memories (visual WM and auditory WM in Fig. 3) before being transmitted to the local processes (motor map in visual area, language-motor-translation, motor map and language-signal-translation in auditory area). The signals generated by those local processes compete to write into the global workspace (see Fig. 3, it consists of three nodes called eye action, hand action and control signal). Each time the content in the global workspace is available, the global processes are, as suggested by the arrows' directions, taking the corresponding content to produce movement trajectory or update the top-down control signal of attention. By these information processes, during the whole life cycle, the agent continuously maintains its inner variables $\{\mathbf{P_{eye}}, \mathbf{P_{hand}}, \mathbf{TDS}\}$, where $\mathbf{P_{eye}}$ indicates positions of eyes gaze, $\mathbf{P_{hand}}$ represents the positions of the hands, and **TDS** is the top-down attention signal. All of them are vectors with size **2**. The details of the model are included in the appendix.

2.2. *Configuration*

Before introducing the data and the training of networks, it is necessary to give a brief view of how the agent is expected to react to different stimuli (Table 1). According to this scheme, the stimuli for the experiments are elaborately generated to make the experiment results distinguishable from each other and analyzable.

As shown, the hand position, eye gaze and the top-down signal are all vectors with a size of 2, corresponding to two hands, two eyes and two modalities, respectively. The default hand position is [0, 0] indicating naturally down and can move to [1, 1] as raise-up. The default position of eyes is [0.5, 0.5] indicating naturally forward gaze, which can move from [0, 0] to [1, 1]. For the top-down control signal, the first element is the competition weight for visual stimuli, and the second is for auditory ones used in the global workspace competition for calculating the relative salience (see appendix for details). Moreover, there are some **None** values in the table, which indicate the corresponding stimulus would not change that variable. Overall, this table does not give the dataset for training, which would be specified in the following training of different networks.

Table 1. Stimuli and corresponding reactions pairs. On each row, the first column is the stimulus the agent attends to; the second column is a description of the agent's reaction, which is reflected in the change of the system variables in the rest columns.

Attended stimulus	Action/inner status	Hand position	Eye gaze	Top-down signal
Blue Object	Lift left hand	[1, 0]	None	None
Grey Object	Lift right hand	[0, 1]	None	None
'Listen to Me'	Enhance auditory stimuli	None	None	[0.5, 1]
Look at Your Right	Look at right side	None	[0.75, 0.75]	None
'Surrender'	Lift both hands; Reset signal; Reset eye gaze	[1, 1]	[0.5, 0.5]	[1, 1]

2.2.1. *Training of motor map process in vision*

The visual environment to the agent is fixed throughout the experiment, displayed in Fig. 4, and the training dataset is given in Fig. 5.

Most of the eight sub-images only contain partly either the blue or gray object, but the most at least occupied the major area. Accordingly, the expected outputs for them are introduced in Table 1. In addition, a convolutional neural network (CNN) [Krizhevsky *et al.*, 2012] is adopted here for image processing (see appendix for details). During the training, the epoch loss is the sum of the whole dataset in a single epoch. The loss for a single sample is calculated by

$$\mathbf{Loss} = \mathbf{Sum}((\mathbf{Output} - \mathbf{Target})^{**}2), \tag{2.1}$$

where **Output** is the actual result of the network, **Target** is the desired prediction and they are vectors with the size of 2. Hence, the loss values are calculated across the whole dataset. Formally

$$\mathbf{Epoch\ Loss} = \sum \mathbf{Loss}_i. \tag{2.2}$$

For the parameter tuning, because the dataset is quite simple for the simulation, only the epoch number and the learning rate **lr** are tested for optimal performance.

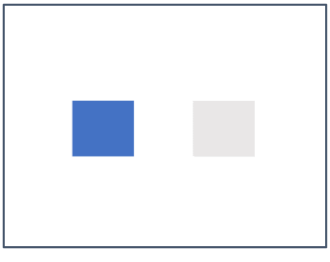

Fig. 4. Visual environment around the agent, which is used as the image input to the agent across all experiments. The agent is attending a certain part of this image according to the bottom-up salience filtering process (see appendix).

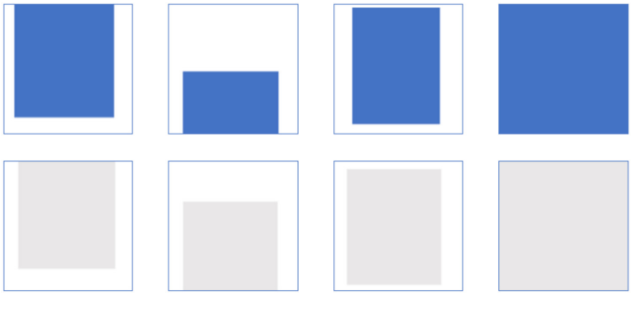

Fig. 5. The whole dataset for the training of the visual motor mapping process. Each of them is a sub-area of Fig. 4 in different scales. The upper four images are regarded as containing the blue object, while the bottom ones are containing the gray object.

With this loss scheme, the network is eventually trained 200 epochs with **lr** of 0.0001 after some preliminary trials.

2.2.2. *Training of local processes in auditory module*

Here, we adopt the same network configuration for the three networks in the auditory module as the input and output formats are the same for them. For the input data to these processes, the speaking data is firstly transformed into image-form spectrums. The visualizations of the spectrums for the three speaking inputs used in the experiment are shown in Fig. 6. This is inspired by the work doing audio classification with CNN models [Kim *et al.*, 2004; Yu and Slotine, 2009; Hershey *et al.*, 2017]. In those studies, the features extracted by short-term Fourier transformation (STFT) guarantee the classification performance on a high level of accuracy.

During the training, the calculation of the epoch loss is the same as (2.1) and (2.2), and the vector size of the output is 2. By this, the network eventually is trained 200 epochs with **lr** of 0.0001 after some preliminary trials.

2.2.3. *Training of eye module and hand module*

As for the action modules, we adopted a recurrent neural network (RNN) [Rumelhart *et al.*, 1986] for the action generation. Gated recurrent unit (GRU) [Cho *et al.*, 2014] is exploited in the hidden layer of the network because the gated mechanism help alleviating the vanishing gradient problem. Similar to auditory processes, the same parameter setting is exploited for the two action processes. The input data for them is in a size of 4, with the first two elements indicating the current position of hands or eye gaze and the last two elements indicating the target position. For the target outputs, which are the predicted movement trajectories of eight steps, they are generated with a fixed movement step size, which is calculated by the following equation:

$$\textbf{Step Size} = (\textbf{Target pose} - \textbf{Current pose})/8. \tag{2.3}$$

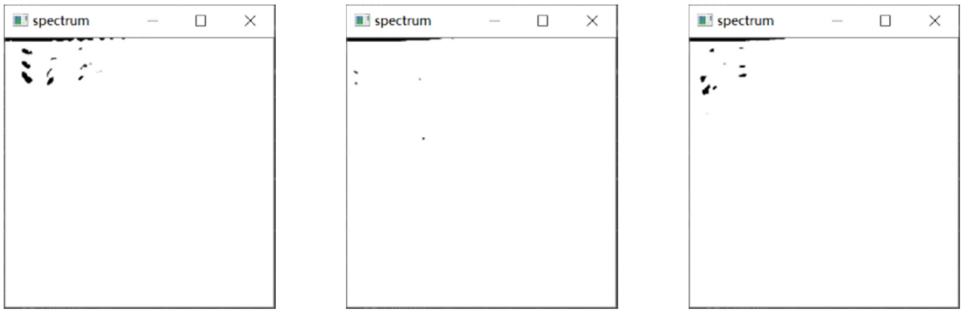

Fig. 6. The whole dataset for the training of auditory processes consists of three language commands, which are recorded manually. These are the spectrums for "listen to me", "look at your right" and "surrender", respectively, from left to right. The transformation is detailed in appendix.

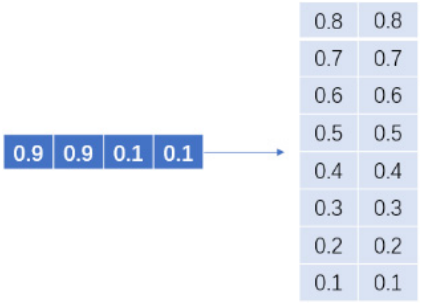

Fig. 7. (Color online) An example of the input vector and output trajectory for the movement module. The dark blue vector is the input vector with [0.9, 0.9] as the initial position, and [0.1, 0.1] as the ending position. The light blue 2D array is the generated trajectory.

Thus, the target trajectory is an array of $\{\mathbf{y_n}|\mathbf{y_n} = \mathbf{current\ pose} + \mathbf{n}^*\mathbf{step\ size}, \mathbf{n} \in (\mathbf{0}, \mathbf{8}]\}$. During the training, the network is expected to learn this pattern of steady movement. The training dataset for these two action processes, 120 pairs of input vectors and output trajectories, are randomly generated. An example is given in Fig. 7.

The calculation of the epoch loss is as same as (2.1) and (2.2), while the output of the network here is a trajectory with the size of $8 * 2$. By this, the network is eventually trained with 2000 epochs with **lr** of 0.0001 after some preliminary trials.

2.3. *Tasks*

After the model implementation and configuration of networks, two experiment tasks were set for the agent to answer the three objective questions discussed in Sec. 1. The first one is to test, with such a GWT-based architecture, the agents' ability to attend to different stimuli with the combined effect of bottom-up and top-down attention. Another one aims to duplicate the attentional blink and lag-1 sparing effects.

2.3.1. *Task 1: Attentional shift*

By this experiment, it is expected to validate the opinion of this work that attention mechanism is a key process for the happening of consciousness, playing as an access gate to consciousness. For this purpose, the agent's ability to shift attention to different stimuli is tested in this task. During the experiment, a sequence of cross-modality stimuli will be input into the agent, thus incurring the movements of hands and eye gaze, and the change of the top-down attention signal. This is feasible as the resulted reactions for different stimuli are unique as displayed in Table 1. To make the experiment understandable to human readers, the task scenario could be described as in Fig. 8.

2.3.2. *Task 2: Attentional blink and Lag-1 sparing*

According to the study of Dehaene *et al.* [2001], while the interval grows, different phenomena would be reported as discussed in Sec. 1. The dynamic pattern over

-The agent is exposed to a visually attractive environment.
-When she is focused on specific object, her parent gently talks to her with: Look at your right. However, she is still focusing on the object without reaction to the language communication;
-Naturally, her parent is becoming angry and shouts out with: Listen to me, which catches her attention, and is interpreted as a top-down control signal to enhance the auditory input in term of salience for competition with visual input;
-Then her parent says again: Look at your right, which enters the agent's attention and be further consciously processed;
-When the eye gaze changes, it incurs the bottom-up filtering process being impacted as the visual focus will automatically change the salience weights of different pixel on the visual input;
-Finally, policemen come as the agent and her parent are suspected as criminals. The policeman shouts out with: Surrender! which breaks into the agent's attention.

Fig. 8. Experiment scenario. Sequentially, each row specifies the stimuli the agent is exposed to and the expected reaction of the agent. All the stimuli are elaborately generated to support this context to happen.

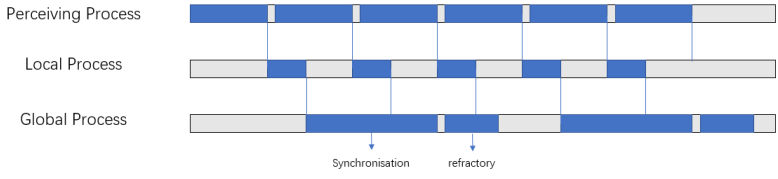

Fig. 9. Temporal dynamics of a three-layer structure. The perceiving process is referred to the salience processing in the modal working memories. In addition, the global workspace is incorporated into the global process in this figure as the working memories are tightly bonded together from the view of consciousness.

time in Fig. 9 could explain these two effects with the system dynamics in an understandable way.

In the figure, there are two periods in the global process, namely the synchronization and the refractory periods. The former is newly proposed in our model to cope with the asynchronization problem, which is attributed to the variants in time costs of different local processes (Table 2). This problem indicates that signals incurred by simultaneously happening stimuli would arrive at the global workspace at close but different times.

To solve this problem, we introduce a synchronization period before the competition phase. By this, when the first signal arrives, there is a delay before the competition. During this period, it is like the agent is processing the stimuli across a short time all together in a competitive way. To ensure this synchronization period to be remarkable, the parameter for this synchronization is set to be **0.3** s in the experiment. The refractory period is referred to the period in which the global process is

Table 2. Computational compositions of different processes in local working memories. The different compositions of salience processing in the two modal working memories are the main reasons for the asynchronization problem.

	Computational compositions
Salience Processing (Vision)	RGB2GreyScale + Pooling + Convolution + Biasing + Max + Mapping
Salience Processing (Audition)	STFT + Max
Motor_map (Vision)	6 * Convolution + 5 * Pooling+Fully connection
Language_signal_translation	
Language_motor_translation	
Motor_map (Audition)	

processing a certain signal. During this period, all newly arriving signals are blocked until it ends. With these two periods, the lag-1 sparing effect will happen when the later stimulus arrives in the global workspace at its synchronization period, and attentional blink will happen during the refractory period.

By simulating the two effects, it is expected to justify the compatibility of the implemented model with the conscious brain of humans.

3. Experiments

3.1. *Attentional shift*

According to the task scenario, at each time step, the agent is exposed to both visual and auditory stimuli. With the change of stimuli, the agent shows different reactions which are reflected by the body positions ($\mathbf{P_{eye}}$, $\mathbf{P_{hand}}$) and inner status (**TDS**). The results are shown in Table 3.

To align with the experiment scenario, the results are interpreted as shown in Fig. 10.

In this experiment, the agent based on the adapted GWT successfully demonstrates its ability to attend to certain stimuli and can shift to various stimuli with the combined effect of bottom-up and top-down mechanisms.

3.2. *Attentional blink and Lag-1 sparing effects*

In this task, the two effects are studied with stimuli in the auditory modality, with a rapid stream of two sound stimuli ("surrender" and "listen to me") as the inputs.

Table 3. Experiment results for the attentional shift. Each row represents the inputs and the resulted reactions at a certain time step, where the image is the fixed visual environment in Fig. 4.

Visual input	Auditory input	Hands pose	Eyes pose	Top-down signal
Image	Look at your right	0.965, 0.002	0.500, 0.500	1.000, 1.000
Image	Listen to me	0.965, 0.002	0.500, 0.500	0.498, 0.996
Image	Look at your right	0.965, 0.002	0.747, 0.748	0.498, 0.996
Image	None	−0.003, 0.994	0.747, 0.748	0.498, 0.996
Image	Surrender	0.994, 1.001	0.494, 0.493	0.996, 0.997

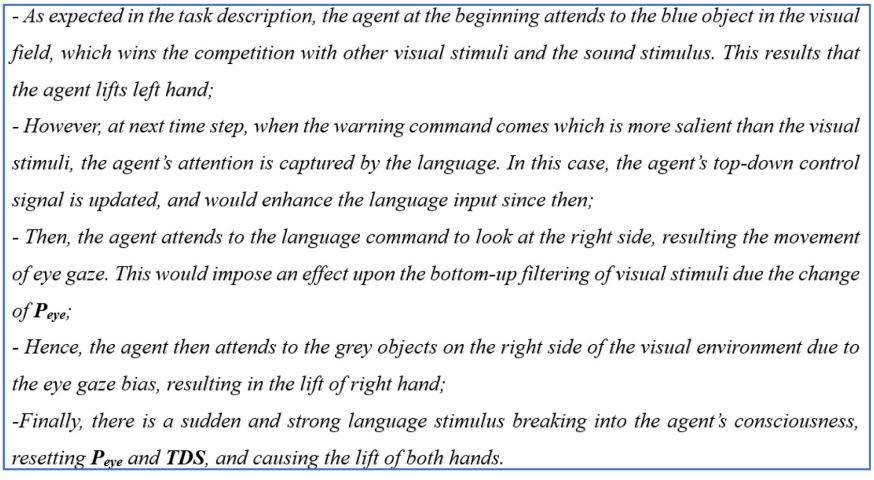

Fig. 10. Experiment results of attention shift. Each row is matched with the experiment scenario in Fig. 8.

To analyze the results, the update process of the top-down attention signal is taken as the analysis subject as the two language inputs will compete for access to it and will result in unique signals on **TDS**. Moreover, for displaying lag-1 sparing effect, the second stimulus arrives during the synchronization period simply takes over the first stimulus instead of competition between them for the demonstrating convenience.

3.2.1. *Trial 1*

The trial is conducted with an increasing interval span between the two stimuli. With each interval value, 10 repetitions are for avoiding occasional bias. When the interval value is large enough for the disappearance of the two effects, the trial ends. The result is displayed in Table 4. The variations within the same row are attributed to the fluctuation of device performance.

Table 4. Results of Trial 1. T1: "Surrender" processed resulting in signal [1, 1], while "listen to me" is arriving within the refractory period hence blocked; T2: "listen to me" arrives during the synchronization period, taking over "surrender", resulting in signal [0.5, 1]; T3: Both inputs are consciously processed, resulting in that the signal firstly becomes [1, 1] and then changes to [0.5, 1].

Rapid stream analysis										
Interval	case 1	case 2	case 3	case 4	case 5	case 6	case 7	case 8	case 9	case 10
0.05	T1	T1	T1	T1	T1	T1	T1	T1	T1	T1
0.10	T1	T1	T1	T1	T2	T1	T2	T2	T1	T2
0.15	T1	T2	T2	T2	T2	T2	T2	T2	T2	T2
0.20	T2	T2	T2	T2	T2	T2	T2	T2	T2	T2
0.25	T2	T2	T3	T2	T3	T2	T2	T2	T2	T2
0.30	T2	T2	T3	T2	T3	T2	T2	T3	T2	T3
0.35	T2	T3	T3	T3	T3	T3	T3	T3	T3	T3

As observed, if the second stimulus is following the first one closely within around 0.1 s, then the second one is remarkably ignored by the agent. This could be attributed to that the modality-specific working memory is occupied for about 0.1 s, which makes itself irresponsive to the latter stimulus. When the interval increases to between 0.1 and 0.25 s, the second stimulus is consciously processed at the cost of abandoning the first one. This is in accordance with the lag-1 sparing feature. Hence, this period could be naturally inferred as the synchronization period of this system. After this period, when the interval is becoming 0.3 s or bigger, the phenomenon fades progressively and the agent consciously perceives both stimuli.

However, the observed dynamics fails to show the attentional blink effect. It is argued that the computations of the global processes in this system, which is inferred as the refractory period, are very simple (eight-step recurrent neural network for actions and no computation for top-down attention update). Hence, the refractory period might be too short to support the observation. To examine this, it is hypothesized that attentional blink is expected to be easily observed if the refractory period is longer (see Trial 2).

3.2.2. *Trial 2*

To test the hypothesis, in this trial, the process of signal update is prolonged by 0.2 s with a processing delay. The experiment result is demonstrated in Table 5.

As expected, the observed results are almost identical to the first trial when the interval between stimuli is less than 0.3 s. In addition, the attentional blink phenomenon is observed when the interval becomes around 0.3−0.5 s, and it fades when the interval gets larger. This also helps determine that the interval between 0.3 and 0.5 s after a stimulus is perceived is the refractory period of this system. In summary, the results displayed above successfully simulate the attentional blink and lag-1 sparing effects, thus verifying the hypothesis concerning the fact that the synchronization and refractory periods of the global processes are responsible for the two effects, respectively.

Table 5. Results of Trial 2. The indications of T1, T2, T3 are the same as in Table 4.

Rapid stream analysis										
Interval	case 1	case 2	case 3	case 4	case 5	case 6	case 7	case 8	case 9	case 10
0.05	T1	T1	T2	T1	T1	T1	T1	T1	T1	T1
0.10	T1	T2	T1	T2	T2	T2	T2	T2	T1	T1
0.15	T2	T2	T2	T2	T2	T2	T2	T2	T2	T2
0.20	T2	T2	T2	T1	T2	T2	T2	T2	T2	T2
0.25	T2	T2	T2	T2	T2	T2	T2	T2	T2	T2
0.30	T2	T1	T1	T1	T1	T1	T2	T2	T1	T1
0.35	T2	T1	T1	T1	T1	T2	T2	T1	T1	T1
0.40	T2	T1	T1	T1	T1	T1	T1	T1	T1	T1
0.45	T1	T1	T1	T1	T1	T1	T1	T1	T1	T1
0.50	T1	T1	T1	T1	T1	T1	T1	T1	T1	T1
0.55	T3	T3	T3	T3	T3	T3	T3	T3	T3	T3

4. Discussion

This work proposes a novel variant of GWT, and integrate the attention mechanism within the model. This model is then validated by two experiments. The first simulation demonstrates the agent with the ability of attentional shift, in which it attends to different stimuli with the combined effect of bottom-up and top-down attentional signals. The second experiment replicates attentional blink and lag-1 sparing, which are two significant phenomena underlying consciousness and attention.

The results of the two experiments directly answer the first two objectives given in Sec. 1. The attentional shift is mainly attributed to the attention mechanism within the model. In this experiment, the bottom-up and top-down attention play equally important roles. Specifically, the underlying causes of the attentional shift are the changes of the perceived stimuli and, or the updates of the top-down attention signal. As shown in the results, the combined effect of these two flows of attention functions as a selective gate to allow only one stimulus, at a time, access the global workspace. This is a sign of entering consciousness as proposed in GWT [Baars, 1993]. Thus, this experiment indicates that implementing the attention mechanism is an inevitable step toward building a conscious agent. For the second experiment, the dynamics of the attentional blink effect is attributed to the distributed and parallel design of modules within our GWT-based framework. Only in such a structure, it reveals the refractory period of the global processes. On the other hand, because of the parallel design, conscious cognition suffers from the asynchronization problem due to computational variants between local processes. The synchronization mechanism is proposed to solve this problem. As a side effect of this, it allows the agent to ignore the temporal order of the stimuli within this period. This exactly accounts for the lag-1 sparing effect, which surprisingly justifies this synchronization mechanism. Although simulating these two effects does not directly help build a conscious system, it is strong evidence supporting the common features between the implemented model and the conscious mind of humans. This is an essential approach for validating theories and models of consciousness.

Compared with the existing work in this area, this work encompasses at least three significances. First, the novelty of separated global workspace nodes in the presented model should be recognized. By this, the GWT is generalized to a wider scope but retain the three key features. It is necessary to reiterate that this adaptation implies a remarkable computation reduction especially in highly complex systems in contrast to the original structure [Baars, 1993; Shanahan, 2006; Dehaene and Changeux, 2011]. Another highlight is the synchronization mechanism which is newly proposed in this work. With this mechanism, both attentional blink and lag-1 sparing effects are successfully simulated. Before this work, only the attentional blink is replicated by Dehaene *et al.* [2003]. In addition to this superficial phenomenon, the synchronization mechanism may imply more about the way the conscious brain cope with the overwhelming information. However, this is beyond the range of the current

discussion. The third emphasis of this work is the integral view toward consciousness. This is in line with the current trend of the research in this area [see, for example, Graziano *et al.*, 2020], which tend to reconcile different theories to account for consciousness. A similar opinion is discussed by Reggia [2013] that consciousness might emerge from the interactions between the specialized processors in the context of GWT. However, the view in this work would resist being equated to that. Alternatively, we emphasize both the development of specialized modules and the interactions between them.

Despite obvious results and the advantages discussed above, we cannot yet claim the emergence of (artificial) consciousness within the presented model. The model is still far from accounting for artificial consciousness from both functional and experiential perspectives [Chalmers, 1996; Reggia, 2013]. Functionally, a conscious model should be capable of or at least provide the foundation for various conscious functions (e.g., short- and long-term memory, emotion, internal world). This problem is usually referred to as the "easy problem" in contrast to the "hard problem" of subjective experience. By such a categorization, it does not imply that implementing those correlates of consciousness is easy, but that one could expect to do that with imaginable approaches [Reggia *et al.*, 2017]. From this perspective, it is evident that the model presented in this work can only account for a very few aspects of conscious functions. While this would be gradually alleviated in the incremental development in the future, the relation between conscious functions and subjective experience is quite harder but essential to ultimately account for the nature of consciousness. The discussion of the contribution of our work to this problem is avoided here, while Reggia *et al.* [2018] propose that solving the computational explanatory gap [Reggia *et al.*, 2014] would get the researchers closer to the hard problem.

Thus, we would like to draw some directions for incremental development in the future based on this work. First, from an engineering perspective, the development of this model has not exploited a complex dataset or interactive environment, which reduces the tension on algorithm design for each computational process. To enhance the work from this, each block inside Fig. 3 could be extended as a narrower research topic, for instance, to achieve more elaborate sensory perceptions and motor ability. From the view of framework design, the developed model is still native from a mature conscious system, which appeals for the implementation of more correlates to consciousness. For instance, within this model, all the networks can only cope with previously learnt (or similar) data, indicating the agent would process novel data in an unpredictably wrong way. In contrast, for a real intelligent agent, it is expected to distinguish learnt and novel information. By this, she could at least avoid the chaos caused by the novel information. Regarding this, the work of Eliasmith [2013] on semantic pointers architecture could provide a feasible approach. An adaptation of this concept into neural network architecture may shed light on the implementation of short- and long-term memory or other knowledge-related modules. Finally, we must establish specific explanations for consciousness itself and the relation between

functions and experience for our model. There are no yet commonly accepted definitions of them, indicating the research across this research community is not well-guided or consistent for all. Though we cannot guarantee a universal definition either, a self-consistent theory of (artificial) consciousness is essential for this research to avoid deviations or unmeaningful repetition.

Acknowledgement

The work of Angelo Cangelosi was partially supported by the following grants: UKRI Trustworthy Autonomous Systems Node in Trust (EP/V026682/1), the US Air Force grant THRIVE++ (FA9550-19-1-7002) and the H2020 e-LADDA ETN (857897).

Appendix A. Model

A.1. *Local working memory*

Though the two working memories are distinct in terms of the pre-processing, nominated information and the computation of the salience **s**, they share common points. In this common structure, the salience value **s** is used to update the top salience **top_s**, which is defined by the highest salience value which the agent has ever received in a certain modality over her whole life. After the calculation and update, respectively, the salience **s** and top salience **top_s** are used in the competition in global workspace (see Appendix A.3). In the following, each working memory is introduced in detail.

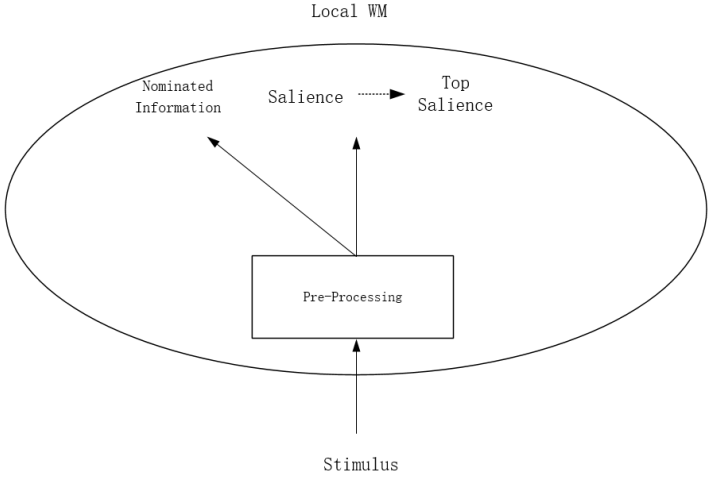

Fig. A.1. Common structure of the local working memory for both modalities. The pre-processing is different for the two modalities, however, both of them generate the nominated information, the candidate competing to exclusively access consciousness, which is a sub-image of the visual environment in the visual module and a transformed spectrum in the auditory module.

A.1.1. *Visual working memory*

Figure A.2 is a visual instantiation of the common structure above. In this modality, the salience of the sub-image is defined by the second derivative of intensity on that area, which is calculated with Laplacian filter kernel [Hummel, 1975], a well-known way for detecting prominent location with a remarkable change in intensity. Figure A.3 shows the kernel we adopted.

Figure A.4 demonstrates the salience filtering process in detail.

The overall purpose of this process is to filter out the most salient part of the image stimulus for later computation and its salience for competition with other stimuli in the auditory module. As illustrated, the original RGB image (**Img**) is converted into a grayscale image by calculating the mean map over the three channels. Formally

$$\mathbf{Img}_{\mathbf{Gray}} = (\mathbf{Img}_{\mathbf{R}} + \mathbf{Img}_{\mathbf{G}} + \mathbf{Img}_{\mathbf{B}})/3, \tag{A.1}$$

where $\mathbf{Img}_{\mathbf{Gray}}$ is the grayscale image and $\mathbf{Img}_{\mathbf{R}}$, $\mathbf{Img}_{\mathbf{G}}$ and $\mathbf{Img}_{\mathbf{B}}$ are three color channels, respectively. Then, $\mathbf{Img}_{\mathbf{Gray}}$ is averagely pooled into three different scales of

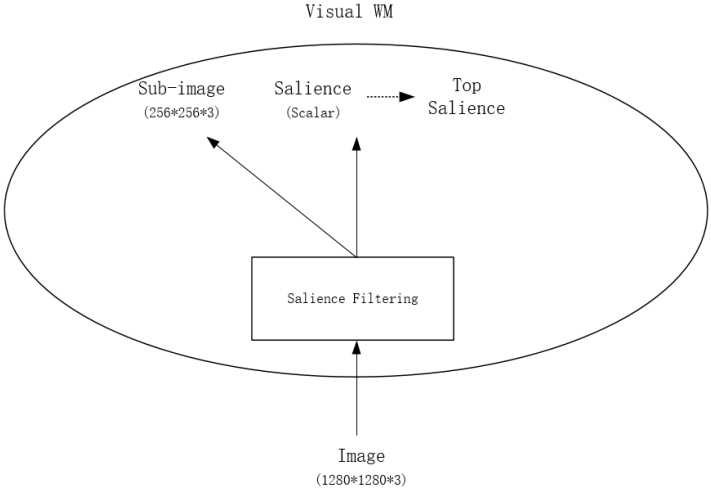

Fig. A.2. Working memory in the visual module. The input image is a matrix of 1280*1280*3 (RGB). The nominated information here is a sub-image of the input image. The size of the sub-image is 256*256*3. The salience filtering process calculates the salience maps on different scales of the input image, then nominates the most salient area.

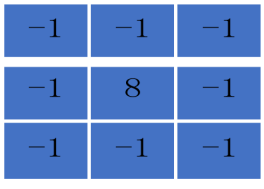

Fig. A.3. Laplacian filter kernel.

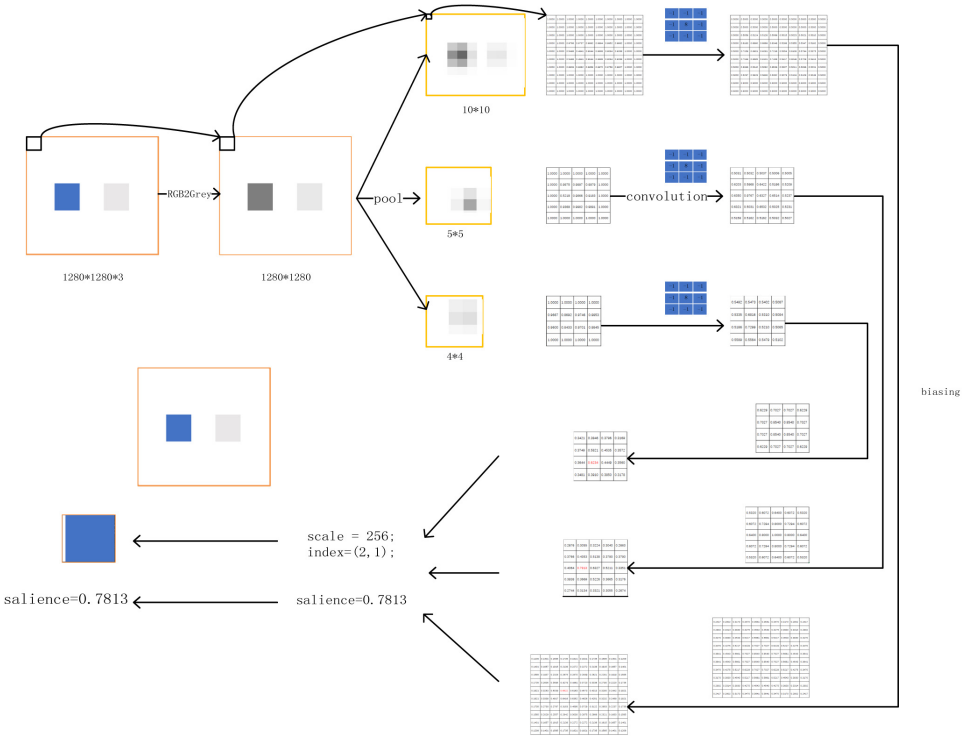

Fig. A.4. Salience filtering process, consisting of RGB2Gray, pooling, convolution, biasing and nominating the salient sub-image with the corresponding salience. Among these, biasing is a multiplication with a bias matrix in element-wise. After this, the three salience maps are used to nominate the most salience area of the input image with the index of the chosen salience.

$\mathbf{10}^*\mathbf{10}(\mathbf{Img_{10}})$, $\mathbf{5}^*\mathbf{5}(\mathbf{Img_5})$ and $\mathbf{4}^*\mathbf{4}(\mathbf{Img_4})$ with the same stride and pooling size of 128, 256, 320, respectively. The scales are picked for simulating attention across multiple scales. This could be extended to any other appropriate scales depends on specific tasks.

In the figure, next to the pooled images, there are three matrices which are their respective intensity distributions. Next, the Laplacian filter is used as the convolution kernel, a $\mathbf{3}^*\mathbf{3}$ matrix as shown in Fig. A.5, with a stride of **1**.

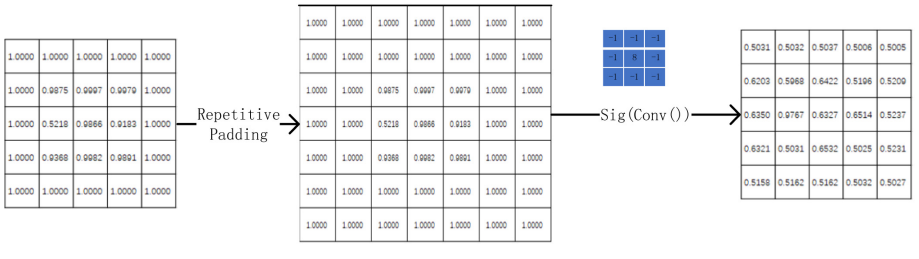

Fig. A.5. Convolution step.

There are two tricks at the beginning and end of this computation. To ensure an element-to-element mapping between the matrices before and after the convolution step, the pooled image is repetitively padded with a size of **1**. Another is that the convolution result is normalized by a sigmoid function. Figure A.5 gives a zoom-in view of this step with $\mathbf{Img_5}$ as an example. Formally:

$$\mathbf{M'_n} = \mathbf{Sig}(\mathbf{Conv}(\mathbf{Pad}(\mathbf{Img_n}))), \tag{A.2}$$

where $\mathbf{M'_n}$ is the salience map of $\mathbf{Img_n}$ before biasing, and n is in 4, 5 and 10 as explained above. By this step, we get three unbiased salience maps $\{\mathbf{M'_4}, \mathbf{M'_5}, \mathbf{M'_{10}}\}$. Then, they are multiplied with the biasing matrices in element-wise. This is for simulating the attentional bias from eye gaze, inspired by the work of Masland [2001], which revealed that the retinal fovea area provides great spatial resolution for the generation of vision and this is not for all the retinal areas. By this, the vision system is more sensitive to the area on the eye gaze position and that progressively decreases with distance away from that position. Hence, the saliences of areas close to the eye gaze position are relatively enhanced. Let the eye gaze position be $[x, y]$ on the visual field, the bias is generated with (A.3)

$$\mathbf{Bias_{ij}} = \mathbf{Pow}(\mathbf{0.8}, \mathbf{Sqrt}((\mathbf{i} - \mathbf{x})^{\mathbf{2}} + (\mathbf{j} - \mathbf{y})^{\mathbf{2}})). \tag{A.3}$$

In the equation, $\mathbf{Bias_{ij}}$ represents the bias value at position $\mathbf{[i, j]}$. 0.8 is the attenuation coefficient, chosen after preliminary trials. By this, the agent is prone to attend to an object closer to the eye gaze position. In Fig. A.4, the eye gaze position is [**0.5**, **0.5**] for two eyes, the bias matrix for $\mathbf{M'_5}$ is shown in Fig. A.6.

With this, we get the three biasing matrices $\{\mathbf{Bias_4}, \mathbf{Bias_5}, \mathbf{Bias_{10}}\}$. The biased salience maps are calculated by

$$\mathbf{M_n} = \mathbf{Bias_n} \odot \mathbf{M_n}, \tag{A.4}$$

where $\odot$ indicates element-wise multiplication, and $\mathbf{n}$ is in 4, 5 and 10.

With the three biased salience maps $\{\mathbf{M_4}, \mathbf{M_5}, \mathbf{M_{10}}\}$, from each, we can get a scale-specific maximum salience value $\mathbf{s_n}$, which is in red in Fig. A.4. Finally, we filtered out the biggest one from the three $\{\mathbf{s_4}, \mathbf{s_5}, \mathbf{s_{10}}\}$ with its **index** and the **scale** it

0.5320	0.6072	0.6400	0.6072	0.5320
0.6072	0.7294	0.8000	0.7294	0.6072
0.6400	0.8000	1.0000	0.8000	0.6400
0.6072	0.7294	0.8000	0.7294	0.6072
0.5320	0.6072	0.6400	0.6072	0.5320

Fig. A.6. Bias matrix for M5′ when eye position is [0.5, 0.5].

belongs to. Formally, the global maximum salience is obtained by

$$\mathbf{s} = \mathbf{Max}(\mathbf{s_n}). \tag{A.5}$$

Moreover, the **index** and **scale** of this global maximum salience are used to calculate the sub-image $\mathbf{Img_{sub}}$ in the original image **Img**, which is characterized by **s**. As illustrated in Fig. A.4, the outputs of this salience filtering process are **s** and $\mathbf{Img_{sub}}$. A reminder, **s** is also used to update the top salience **top_s** as explained at the beginning of this section. However, the size of $\mathbf{Img_{sub}}$ is not guaranteed to be **256*256*3** each time as it is determined by the scale to which **s** belongs. Hence, for the convenience of later computation, $\mathbf{Img_{sub}}$ will be resized into **256*256*3** and stored in the working memory along with **s**.

A.1.2. *Auditory working memory*

In Fig. A.7, the speaking input **R** here is a vector of **12,000** samples (duration is 2 s with 6000 frames per second), each of which is a scalar number. The nominated information here is a two-dimensional (2D) spectrum **F** of the speaking input. The size of the spectrum is **256*256**. The salience **s** of this modality is defined by the maximum value of the original speaking vector. Formally

$$\mathbf{s} = \mathbf{Max}(\mathbf{R}). \tag{A.6}$$

The top salience here is exactly as same as defined at the beginning of this section. Different from the vision, there is no competition between different stimuli inside the auditory module, but only a spectrum transformation with salience calculation.

In the transformation, the speaking data **R** is firstly transformed into a time-frequency data $\mathbf{F}'$ by STFT with the **librosa** library in Python. Namely, the x-axis is the time-space and the y-axis is the frequency space. $\mathbf{F}'$ is a matrix of complex

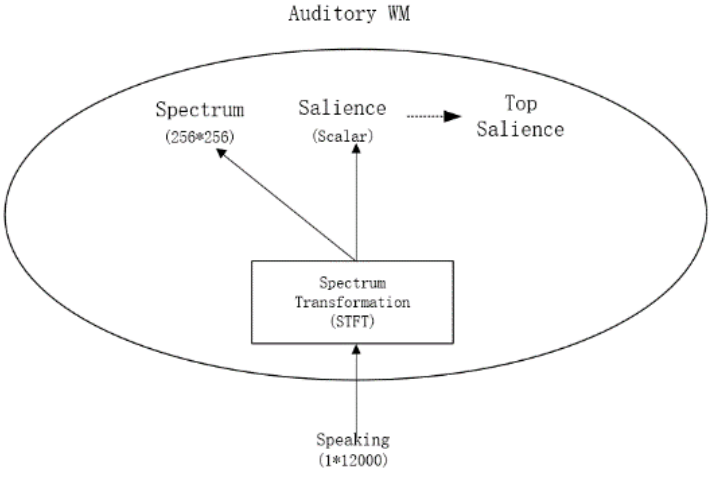

Fig. A.7. Auditory working memory.

numbers, and the real and imaginary parts represent the magnitude and phase of the frequency, respectively. Furthermore, for implementation, $\mathbf{F}'$ is then transformed into a matrix of decibels, the desired spectrum $\mathbf{F}$, which is calculated by

$$\mathbf{F} = \mathbf{mag2db}\ (\mathbf{log}(\mathbf{abs}(\mathbf{F}'))), \tag{A.7}$$

where **abs** function is extracting the magnitude, which is the real part, of the elements of $\mathbf{F}'$, **log** is for number scale control and **mag2db** is the function transforming magnitude values into decibels. The resulted spectrum $\mathbf{F}$ of (A.7) is a $\mathbf{151}^*\mathbf{161}$ matrix, which is resized into $\mathbf{256}^*\mathbf{256}$ for the convenience of later processing. Finally, the resized spectrum $\mathbf{F}$ along with the salience $\mathbf{s}$ calculated by (A.6) are stored in the working memory for later processes.

A.2. *Local processes*

In this model, the visual module has only one local process (motor map) and the auditory module has three (motor map, language-motor-translation and language-signal-translation). We do not focus on the psychological mechanisms behind them, rather they are designed for providing signals to different global processes. Generally, the motor map processes in both modalities are generating signals for hand action, and language-motor-translation and language-signal-translation are for eye action and top-down signal update, respectively.

Notably, all of the four local processes are implemented with the CNN model due to the sub-image $\mathbf{Img_{sub}}$ and spectrum $\mathbf{F}$ generated by the two working memories are 3D and 2D data, which contains rich spatial features. Hence, we can exploit a quite similar network structure for all four processes (Fig. A.8). The structure consists of six blocks of convolution and a linear mapping layer. Each convolutional block contains a repetitive padding of size **1**, a convolution layer with a kernel size of $\mathbf{3}^*\mathbf{3}$ and an average pooling of $\mathbf{2}^*\mathbf{2}$ (the final block does not have pooling). The numbers of convolution kernels for each block are directly reflected by the resulted number of channels in the figure, which are $\{16, 32, 64, 32, 16, 8\}$. Formally, the computation of a convolution block is described by

$$\mathbf{Block}(\mathbf{X}) = \mathbf{Pool}(\mathbf{Conv}(\mathbf{Pad}(\mathbf{X}))), \tag{A.8}$$

where $\mathbf{X}$ is the input into that block. Finally, the result of the final convolution block is a matrix of $\mathbf{8}^*\mathbf{8}^*\mathbf{8}$, which is converted into a vector with a size of **512** and then passes a linear mapping layer. The output of the model is a vector with a size of **2**, which has different definitions in different processes. Also, the inputs are different for the two modalities, which are further explained in the following.

A.2.1. *Motor map in the visual module*

The input of the instantiation is the sub-image $\mathbf{Img_{sub}}$ ($\mathbf{256}^*\mathbf{256}^*\mathbf{3}$) generated by the visual working memory. The output [**a**, **b**] here is the target position for hand movement with two hands.

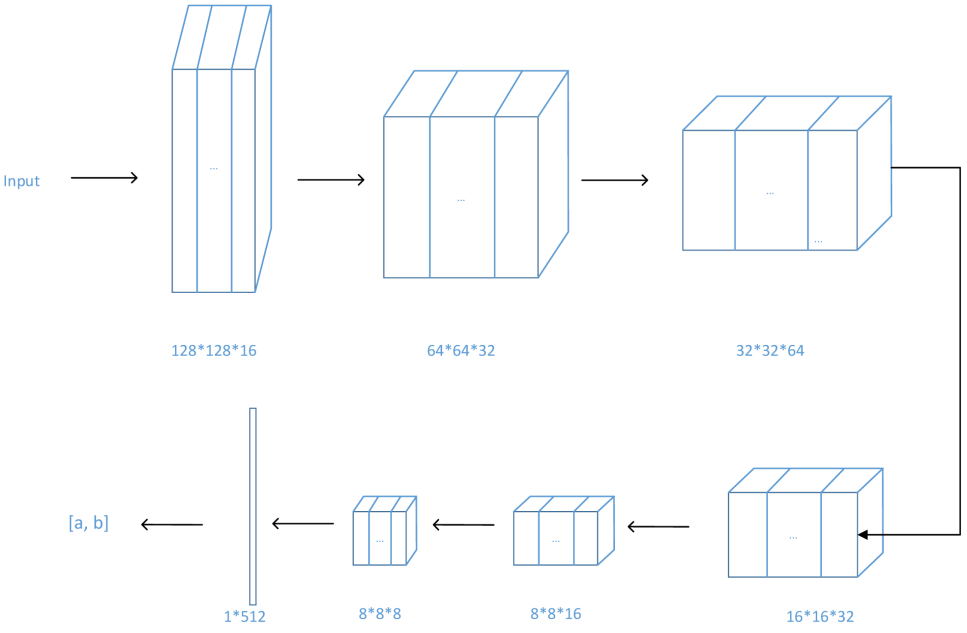

Fig. A.8. CNN model for the local processes.

A.2.2. *Motor map in the auditory module*

The input for this process is the spectrum **F** (**256*****256**) generated by the auditory working memory. The output [**a**, **b**] here is as well the target position for hand movement.

A.2.3. *Language-motor-translation in auditory module*

The input for this process is the spectrum **F** (**256*****256**) generated by the auditory working memory. The output [**a**, **b**] here is the target position for eye gaze movement with two eyes.

A.2.4. *Language-signal-translation in auditory module*

The input for this process is the spectrum **F** (**256*****256**) generated by the auditory working memory. The output [**a**, **b**] here is the target status for top-down control signal corresponding to visual and auditory modalities, respectively.

With the computations explained above, at a certain time, there are two handles, the visual $\mathbf{Comb_v}$ and the auditory $\mathbf{Comb_a}$, respectively. Each of them is a combination of $\{\mathbf{H}, \mathbf{E}, \mathbf{C}\}$, which are targets for hand movement, eye movement and top-down control signal generated by the above processes. Notably, **E** and **C** in $\mathbf{Comb_v}$ are always **None** as no process in vision generating these two signals in our model. Similarly, if there are no corresponding signals generated, the content is **None** at a certain time.

So far, the computations in local areas are all covered. It is better to give a summary of the generated ingredients for later computation in Global Workspace

and global areas. We use the notations $\mathbf{L_v}$ and $\mathbf{L_a}$ for the set of those ingredients from two modalities. Formally:

$$\mathbf{L_v} = \{\mathbf{Comb_v}, \mathbf{s}, \mathbf{top\_s}\}, \tag{A.9}$$

$$\mathbf{L_a} = \{\mathbf{Comb_a}, \mathbf{s}, \mathbf{top\_s}\}. \tag{A.10}$$

In the two equations, **s** and **top_s** are from respective local working memories. Since then, all the later processes only involve the elements of $\mathbf{L_v}$ and $\mathbf{L_a}$.

A.3. *Global workspace*

The global workspace contains a cache **G** of three nodes, and there are two mechanisms implemented underlying global workspace. One is synchronization and another is competition. The activity is formally described as shown in Fig. A.9.

A.3.1. *Synchronization*

Due to the variants in time costs of different local processes, the signals incurred by simultaneously happening stimuli would arrive at the global workspace at close but different times. As shown in Table 2 and explanation of those processes in previous parts, the local processes of both modalities share the same structure, hence it should contribute nothing or very little to the asynchronization. Conversely, the different compositions of salience processing in the two modal working memories could be the main reason for that phenomenon mentioned above. Thus, we introduce a synchronization period before the next competition phase. By this, when the first signal arrives, there is a delay before the competition. During this period, it is like the agent is processing the stimuli across a short time all together in a competitive way. The parameter for this synchronization is investigated in the experiment part.

A.3.2. *Competition*

The generated signals $\mathbf{Comb_v}$ and $\mathbf{Comb_a}$ from the local processes detailed in Appendix A.2 are not always guaranteed to be consciously processed in the global area. The global activities are determined by the competition in global workspace.

```
Loop:
    if all elements in Combv and Comba are None:
        Continue;
    else:
        Synchronisation;
        Competition;
        Clear Combv, Comba, Lv[s], La[s];
```

Fig. A.9. Recurrent activity of global workspace.

Cache **G** is corresponding to the content of $\mathbf{Comb_v}$ and $\mathbf{Comb_a}$. Both of these two sets will try to write into the cache. This incurs a competition between two modalities with respect of the stimuli saliences $\mathbf{L_v}[\mathbf{s}]$ and $\mathbf{L_a}[\mathbf{s}]$. However, they are not directly ready for the cross-modality competition due to the scale variants. Hence, we introduce relative saliences $\mathbf{s_v}$ and $\mathbf{s_a}$ for competition. Formally:

$$\mathbf{s_v} = \mathbf{L_v}[\mathbf{s}]/\mathbf{L_v}[\mathbf{top\_s}], \tag{A.11}$$

$$\mathbf{s_a} = \mathbf{L_a}[\mathbf{s}]/\mathbf{L_a}[\mathbf{top\_s}]. \tag{A.12}$$

According to (A.11) and (A.12), the $\mathbf{s_v}$ and $\mathbf{s_a}$ are normalized into a comparable scale. The competition is illustrated as shown in Fig. A.10.

TDS is the maintained top-down control signal. Hence, **G** is also in form of {**H**, **E**, **C**}.

Figure A.11 is an example showing the competition between an image stimulus and a speaking stimulus, recapping the overall architecture. The left part represents the flow of signals from sensory inputs to global workspace content **G**, and the right part is the exploitations of the **G** in the global areas. The dotted lines indicate that there is no real signal transmission but a theoretical connection. In this example, $\mathbf{P_{eye}}$ and **TDS** are at the defaults. The visual and auditory circles here include the modal working memory and the corresponding local processes, respectively. Hence, with the fixed visual environment, the filtered sub-image $\mathbf{Img_{sub}}$ and the salience **s** are exactly as illustrated in Fig. A.4.

Then, $\mathbf{Img_{sub}}$ is input into the local process, which generates a signal [1, 0] corresponding to hand movement. Because there is no process in vision generating

```
if sv * TDS [0] is equal to or greater than sa * TDS [1]:
    G = Combv;
else:
    G = Comba;
```

Fig. A.10. Competition process.

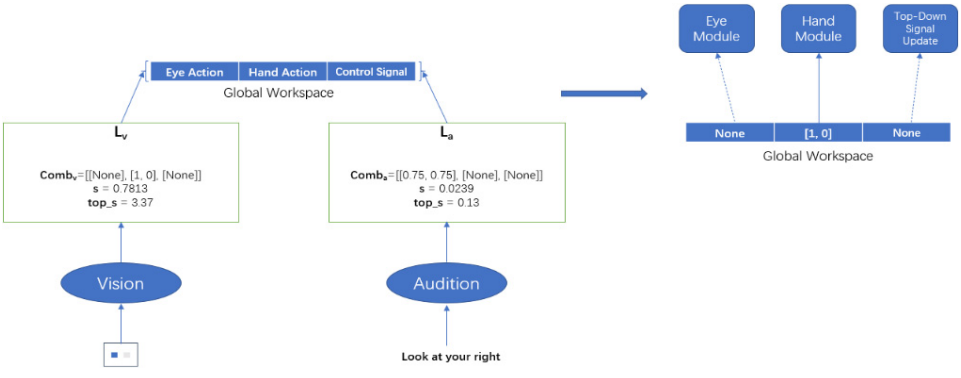

Fig. A.11. Competition with Peye = [0.5, 0.5], TDS = [1, 1].

signals for eye movement and top-down control, $\mathbf{Comb_v}$ is as illustrated in the figure. For the auditory module, the input is the language command "**Look at your right**". Similarly, the corresponding spectrum $\mathbf{F}$ and salience $\mathbf{s}$ are calculated. The resulted $\mathbf{Comb_a}$ only contains a valid signal for eye movement. The top saliences shown in the figure for the two modalities are the values at a certain time. Next, based on the saliences and top saliences from both modalities, according to (A.11) and (A.12), the relative saliences $\mathbf{s_v} = 0.23$ and $\mathbf{s_a} = 0.18$ are computed, respectively. This results in the winning of vision, and the global workspace $\mathbf{G}$ is updated by $\mathbf{Comb_v}$ as illustrated. After this, the global workspace content $\mathbf{G}$ is then transmitted to the corresponding global modules, which are further introduced in the next section.

A.4. *Global processes*

There are three global processes, eye module, hand module and top-down control signal update. The eye and hand modules are realized with action generation processes based on RNN [Rumelhart *et al.*, 1986], which contains a linear input layer, a hidden layer of GRU [Cho *et al.*, 2014] cell and a linear output layer as demonstrated in Fig. A.12. The circle arrow pointing back to the GRU cell indicates the recurrent forwarding of hidden states. The number of steps is 8.

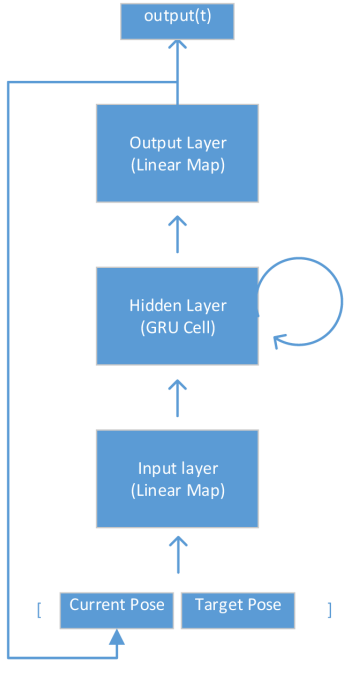

Fig. A.12. Recurrent neural network for action processes.

For each step, the output is formally calculated by

$$\mathbf{output_t} = \mathbf{Linear_{out}}(\mathbf{GRU}(\mathbf{Linear_{in}}(\mathbf{X_t}))) \tag{A.13}$$

and

$$\begin{aligned} &\mathbf{X_t} = \text{concatenate } (\mathbf{P_c}, \mathbf{T}) \text{ if } \mathbf{t} = \mathbf{0}; \\ &\text{otherwise } \mathbf{X_t} = \text{concatenate } (\mathbf{output_{t-1}}, \mathbf{T}), \end{aligned} \tag{A.14}$$

where $\mathbf{P_c}$ is the current position and $\mathbf{T}$ is the target position for movement, which are both vectors with a size of **2**. The two vectors are combined into a vector with a size **4** as the input $\mathbf{X_t}$, and for $\mathbf{t} > \mathbf{0}$, $\mathbf{P_c}$ is replaced by the output of the previous step $\mathbf{output_{t-1}}$ according to (A.14). The output at each step is a vector with a size of **2**. Hence, the result of this RNN model is finally a movement trajectory **MT** with a size of **8*****2**. Formally:

$$\mathbf{MT} = \{\mathbf{output_t} | \mathbf{t} \in [0, 7]\} = \mathbf{RNN} \quad (\mathbf{P_c}, \mathbf{T}). \tag{A.15}$$

Furthermore, the hidden units are set to be **100**, which indicates that the weight matrices of the input layer, GRU cell and output layer are **4*****100**, **100*****100** and **100*****2**, respectively. To be connected with the computation of global workspace, the three global processes are further detailed as follows.

A.4.1. *Eye module*

In this instantiation of the model, $\mathbf{T} = \mathbf{G}[\mathbf{E}]$ and $\mathbf{P_c}$ is read from the inner variable $\mathbf{P_{eye}}$. Hence, the movement trajectory for the eye module is generated by

$$\mathbf{MT_{eye}} = \mathbf{RNN} \quad (\mathbf{P_{eye}}, \mathbf{G}[\mathbf{E}]). \tag{A.16}$$

The $\mathbf{P_{eye}}$ is then updated by $\mathbf{MT_{eye}}[-1]$.

A.4.2. *Hand module*

For the hand module, $\mathbf{T}$=$\mathbf{G}[\mathbf{H}]$ and $\mathbf{P_c}$ is read from the inner variable $\mathbf{P_{hand}}$. Hence, the movement trajectory for the hand module is generated by

$$\mathbf{MT_{hand}} = \mathbf{RNN}(\mathbf{P_{hand}}, \mathbf{G}[\mathbf{H}]). \tag{A.17}$$

The $\mathbf{P_{hand}}$ is then updated by $\mathbf{MT_{hand}}[-1]$.

A.4.3. *Top-down control signal update*

The top-down signal update process is only a overwrite to the agent's inner variable of the top-down control signal without any computation. So simply **TDS** is directly replaced by $\mathbf{G}[\mathbf{C}]$.

A.5. *Summary*

So far, all the computation processes within this model are introduced. It is clear to conclude that the agent is perceiving the environment visually and auditorily with the combined effects of the bottom-up filtering and top-down control signal. The inner states of the agent play significant roles (e.g., eye gaze $\mathbf{P}_{\mathbf{eye}}$ and **TDS**). Furthermore, the agent, during its life cycle, continuously imposes effects on and maintains those inner states with those computational processes.

References

Baars, B. J. [1993] *A Cognitive Theory of Consciousness* (Cambridge University Press).

Baars, B. J. [2002] The conscious access hypothesis: Origins and recent evidence, *Trends Cogn. Sci.* **6**(1), 47–52.

Baars, B. J. and Franklin, S. [2009] Consciousness is computational: The LIDA model of global workspace theory, *Int. J. Mach. Conscious.* **1**(1), 23–32.

Blum, M. and Blum, L. [2021] A theoretical computer science perspective on consciousness, *J. Artif. Intell. Conscious.* **8**(1), 1–42.

Buschman, T. J. and Miller, E. K. [2007] Top-down versus bottom-up control of attention in the prefrontal and posterior parietal cortices, *Science* **315**(5820), 1860–1862.

Chalmers, D. J. [1996] *The Conscious Mind: In Search of a Fundamental Theory* (Oxford Paperbacks).

Chella, A. and Manzotti, R. [2007] "Artificial intelligence and consciousness," in *Association for the Advancement of Artificial Intelligence Fall Symp.*, pp. 1–8.

Chella, A. and Manzotti, R. [2011] "Artificial consciousness," in *Perception-Action Cycle* (Springer, New York, NY), pp. 637–671.

Chella, A. and Manzotti, R. [2013] *Artificial Consciousness* (Andrews UK Limited).

Chella, A., Cangelosi, A., Metta, G. and Bringsjord, S. [2019] Consciousness in humanoid robots, *Front. Robot. AI* **6**, 17.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H. and Bengio, Y. [2014] Learning phrase representations using RNN encoder-decoder for statistical machine translation, arXiv:1406.1078.

Dehaene, S. and Changeux, J. P. [2011] Experimental and theoretical approaches to conscious processing, *Neuron* **70**(2), 200–227.

Dehaene, S., Naccache, L., Cohen, L., Le Bihan, D., Mangin, J. F., Poline, J. B. and Rivière, D. [2001] Cerebral mechanisms of word masking and unconscious repetition priming, *Nat. Neurosci.* **4**(7), 752–758.

Dehaene, S., Sergent, C. and Changeux, J. P. [2003] A neuronal network model linking subjective reports and objective physiological data during conscious perception, *Proc. Natl. Acad. Sci.* **100**(14), 8520–8525.

Franklin, S. [2003] A conscious artifact?, *J. Conscious. Stud.* **10**(4–5), 47–66.

Goertzel, B. [2007] *Artificial General Intelligence*, C. Pennachin (ed.), Vol. 2 (Springer, New York).

Graziano, M. S., Guterstam, A., Bio, B. J. and Wilterson, A. I. [2020] Toward a standard model of consciousness: Reconciling the attention schema, global workspace, higher-order thought, and illusionist theories, *Cogn. Neuropsychol.* **37**(3–4), 155–172.

Haikonen, P. O. [2003] *The Cognitive Approach to Conscious Machines* (Imprint Academic).

Hershey, S., Chaudhuri, S., Ellis, D. P., Gemmeke, J. F., Jansen, A., Moore, C., Plakal, M., Platt, D., Saurous, R. A., Seybold, B., Slaney, M., Weiss, R. and Wilson, K. [2017] "CNN

architectures for large-scale audio classification," *2017 IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)* (IEEE), pp. 131–135.

Holland, O. and Goodman, R. [2003] Robots with internal models a route to machine consciousness?, *J. Conscious. Stud.* **10**(4–5), 77–109.

Hommel, B. and Akyürek, E. G. [2005] Lag-1 sparing in the attentional blink: Benefits and costs of integrating two events into a single episode, *Q. J. Exp. Psychol. A* **58**(8), 1415–1433.

Kim, H. G., Moreau, N. and Sikora, T. [2004] Audio classification based on MPEG-7 spectral basis representations, *IEEE Trans. Circuits Syst. Video Technol.* **14**(5), 716–725.

Knudsen, E. I. [2007] Fundamental components of attention, *Annu. Rev. Neurosci.* **30**, 57–78.

Krizhevsky, A., Sutskever, I. and Hinton, G. E. [2012] Imagenet classification with deep convolutional neural networks, *Adv. Neural Inf. Process. Syst.* **25**, 1097–1105.

Manzotti, R. and Chella, A. [2020] Conscious machines: A possibility? If so, how?, *J. Artif. Intell. Conscious.* **7**(2), 183–198.

Masland, R. H. [2001] The fundamental plan of the retina, *Nat. Neurosci.* **4**(9), 877–886.

Raymond, J. E., Shapiro, K. L. and Arnell, K. M. [1992] Temporary suppression of visual processing in an RSVP task: An attentional blink?, *J. Exp. Psychol. Hum. Percept. Perform.* **18**(3), 849.

Reggia, J. A. [2013] The rise of machine consciousness: Studying consciousness with computational models, *Neural Netw.* **44**, 112–131.

Reggia, J. A., Huang, D. W. and Katz, G. [2017] Exploring the computational explanatory gap, *Philosophies* **2**(1), 5.

Reggia, J. A., Katz, G. E. and Davis, G. P. [2018] Humanoid cognitive robots that learn by imitating: Implications for consciousness studies, *Front. Robotics AI* **5**, 1.

Reggia, J. A., Katz, G. E. and Davis, G. P. [2020] Artificial conscious intelligence, *J. Artif. Intell. Conscious.* **7**(1), 95–107.

Reggia, J. A., Monner, D. and Sylvester, J. [2014] The computational explanatory gap, *J. Conscious. Stud.* **21**(9–10), 153–178.

Rumelhart, D. E., Hinton, G. E. and Williams, R. J. [1986] Learning representations by back-propagating errors, *Nature* **323**(6088), 533–536.

Shanahan, M. [2006] A cognitive architecture that combines internal simulation with a global workspace, *Conscious. Cogn.* **15**(2), 433–449.

Shanahan, M. [2012] The brain's connective core and its role in animal cognition, *Philos. Trans. R. Soc. B, Biol. Sci.* **367**(1603), 2704–2714.

Shanahan, M. and Connor, D. [2008] "Modeling the neural basis of cognitive integration and consciousness," *ALIFE*, pp. 553–560.

Starzyk, J. A. and Prasad, D. K. [2011] A computational model of machine consciousness, *Int. J. Mach. Conscious.* **3**(2), 255–281.

Taylor, J. G. [2007] CODAM: A neural network model of consciousness, *Neural Netw.* **20**(9), 983–992.

Tinsley, C. J. [2008] Using topographic networks to build a representation of consciousness, *Biosystems* **92**(1), 29–41.

Tsuchiya, N. and Adolphs, R. [2007] Emotion and consciousness, *Trends Cogn. Sci.* **11**(4), 158–167.

Vimal, R. [2009] Meanings attributed to the term 'consciousness': An overview, *J. Conscious. Stud.* **16**(5), 9–27.

Yu, G. and Slotine, J. J. [2009] "Audio classification from time-frequency texture," *2009 IEEE Int. Conf. Acoustics, Speech and Signal Processing (IEEE)*, pp. 1677–1680.