Emergence of Content-Agnostic Information Processing by a Robot Using Active Inference, Visual Attention, Working Memory, and Planning

Accepted for publication in **Neural Computation**, Mar 18, 2021

Jeffrey Frederic Queißer[a], Minju Jung[b], Takazumi Matsumoto[a], Jun Tani[a,*]

[a] *Okinawa Institute of Science and Technology, Okinawa, Japan*
[b] *Brown University, Providence, RI, USA*

Abstract

Generalization by learning is an essential cognitive competency for humans. For example, we can manipulate even unfamiliar objects and we can generate mental images before enacting a preplan. How is this possible? The current study investigated this problem by revisiting our previous study (Jung et al., 2019), which examined the problem of vision-based, goal-directed planning by robots performing a task of block stacking. By extending the previous study, the current work introduces a large network comprised of dynamically interacting sub-modules, including visual working memory (VWMs), a visual attention module, and an executive network. The executive network predicts motor signals, visual images, and various controls for attention, as well as masking of visual information. The most significant difference from the previous study is that the current model contains an additional VWM. The whole network is trained by using predictive coding and an optimal visuo-motor plan to achieve a given goal state is inferred using active inference. Results indicate that the current model performs significantly better than that used in Jung et al. (2019), especially when manipulating blocks with unlearned colors and textures. Simulation results revealed that the observed generalization was achieved because *content-agnostic* information processing developed through synergistic interaction between the 2nd VWM and other modules during the course of learning, in which memorizing image contents and transforming them are dissociated. This paper verifies this claim by conducting both qualitative and quantitative analysis of simulation results.

Keywords: visual working memory; goal directed planning; active inference; predictive coding; variational bayes; recurrent neural network

*The correspondence should be sent to tani1216jp@gmail.com

1. Introduction

How can artificial agents, as well as humans, acquire knowledge and skills necessary to generate goal-directed action using complex sensory streams such as vision, with generalization? Specifically, how can they deal with unlearned situations as human do? If complex, goal-directed action generation requires adequate communication and arbitration among different higher cognitive processes, such as plan generation, attention, and working memory, how can they develop autonomously? The current study examines these questions by conducting a synthetic modeling study using a robotic experimental platform. First, we consider what sorts of higher cognitive competencies in robots would be crucial for reconstruction of human cognitive behaviors.

1.1. Higher cognitive competencies

One of the most essential higher cognitive competencies for humans is the ability to learn to develop internal models of the world through iterative interactions with it (Wolpert et al., 1998). Learning of the internal model must involve extracting latent structure from partially observed sensory streams that could involve uncertainty and require probabilistic representations, as required in many real world situations. Acquired internal models can be used for various types of cognitive processes, such as imaging possible future outcomes (Jeannerod, 1994) or rehearsing sensory-motor events experienced in the past (Epstein, 1980). In addition, internal models can be used for various inferences, such as the current state, from sensory inputs or optimal action plans to achieve desired goals by incorporating the aforementioned mental processes (Friston, 2013). Related to this, it has been suggested that humans are capable of extracting causal rules from repeated observations of physical phenomena. Developmental psychologists have shown that human infants acquire basic physical causal rules in early development (Reddy, 2008). For example, they learn that when they act on an object, the appearance of the object could change, or that it will remain the same if untouched, an attribute known as object permanency (Baillargeon et al., 1985).

Another essential higher cognitive competency is compositionality, by which the whole can be composed/decomposed into reusable parts (Evans, 1982). Although the idea of compositionality comes originally from language, this also accounts for other modalities such as those involving vision or proprioception. Visual systems in humans and other animals develop compositional representation for complex visual objects in hierarchically organized visual pathways (Van Essen & Maunsell, 1983; Tanaka, 1996). Likewise, it has been widely assumed that various complex actions can be flexibly generated by adequate re-composition with a set of behavior primitives (Arbib, 1981) using hierarchical information processing (Rosenbaum, 1991; Fuster, 2004). Here, it is crucial to consider that the objective of learning is not just to remember exact experiences of the past in the manner of a video recorder, but to extract essential compositional structures along with hierarchical organization. This is the way to gain generalization in representing skills and knowledge acquired through limited sensory-motor experience. Various related computational models have been proposed for vision (Fukushima & Miyake, 1982; Weng et al., 1993) and for action (Kuniyoshi et al., 1994; Yamashita & Jun, 2008).

The competency for attention and effective use of working memory must be also crucial. Humans can attend to an important part of bulk information flow and can segment it from the background using top-down prior knowledge (Posner, 1995). Then, segmented information is often saved in working memory for further manipulation using other information (Luck & Vogel, 1997; Downing, 2000). The information to be attended and manipulated using working memory may be abstracted at a higher level, such as in the prefrontal cortex (Fuster, 2015; Goldman-Rakic, 1995) as well as lower sensory signals, such as from vision and audition (Harrison & Tong, 2009; Nyberg et al., 2000; Kumar et al., 2016).

Furthermore, humans have cognitive competency by which they can generalize experiences in familiar situations to those in unfamiliar situations (Saffran et al., 1996; McClelland & Plaut, 1999). As an example, humans can physically and mentally manipulate not only familiar objects, but also those having novel features, such as shape, size, and color. We can grasp and lift a novel mug and can also image this action without much difficulty. Surprisingly, humans can achieve this sort of generalization with only limited experience. How is this possible? The current paper focuses especially on this question, as will be detailed later. Although there are undoubtedly other higher cognitive competencies essential to human cognition, such as social cognitive capability, the current study focuses on those mentioned above.

1.2. Development of cognitive competencies via synergy

Higher cognitive competencies required for different aspects of human-like cognitive processes raise interesting questions. How do they develop, and how can each of them be adequately coordinated with others to maximize performance of the whole in solving various cognitive tasks? It would be reasonable to presume that a single neural network cannot produce such a coordinated assembly of cognitive competencies, but it could conceivably develop in a dynamic network allowing synergistic interactions among inter-connected sub-modules. Furthermore, the function of individual sub-module networks may not be programmed by evolution, but instead they may be self-organized through synaptic plasticity in the course of learning to interact with other sub-modules, as neurodevelopmental studies (Sur & Rubenstein, 2005; Rakic, 2009; Li et al., 2016) suggest.

There have been numerous studies to build an integrative brain model consisting of mutually interacting sub-module networks. Since the middle 1990s, O'Reilly and colleagues (O'Reilly, 2006; O'Reilly & Frank, 2006) developed the so-called Leabra Cognitive Architecture to simulate an integrative brain model using a connectionist approach. The integrative brain model consists of sensory and motor inputs/outputs, the prefrontal cortex (PFC), the posterior cortex, the basal ganglia (BG), and the thalamus. The model explains well a mechanism for higher cognitive function assumed in the PFC in terms of dynamic gating of working memory in the PFC by the basal ganglia. However, their models involve neither predictive internal models nor temporal processes.

For more than 2 decades, Edelman and colleagues (Edelman, 1993) have developed integrative brain models based on the theory of *Neural Darwinism* using a series of DARWIN robots. This theory postulates that variation and selection within neural populations drive development and function of the brain. The latest version, DARWIN X (Krichmar et al., 2005), was developed to investigate the problem of spatial memory development in rodents.

The simulated neural network model comprised 90,000 neural units in 50 brain areas including a visual system, a head direction system, a hippocampal formation, a basal forebrain, a value or reward system, and an action selection system. The navigation learning experiment using the model showed a nontrivial result that placed cell-like structures developed in the CA1 region in the model network just by providing biologically plausible connectivity with other regions. This embodied integrative brain model study postulates that brain function in each brain region can develop through postnatal sensory-motor experiences by utilizing anatomical connectivity between brain regions. This model, however, does not deal with human-level higher cognitive competency, such as goal-directed planning using learned internal models.

Eliasmith and colleagues (Eliasmith et al., 2012; Eliasmith, 2013) developed the so-called Neural Engineering Framework, which can generate neural systems consisting of millions of spiking neurons allocated to more than 20 different brain regions, including both cortical and sub-cortical areas. The neural system demonstrates a set of impressive higher cognitive tasks, including serial working memory tasks, questions, and answers, and fluid reasoning between inputs/outputs relation. However, the mechanism is devised in a purely engineering way, using a sort of neural compiler with a powerful parameter-setting mechanism for determining optimal synaptic weights. Therefore, it would be difficult for the model to acquire organizing principles to develop higher cognitive mechanisms based on learning of sensory-motor experience.

1.3. Our prior study and new trials in the current study

Although the aforementioned studies have many interesting features, they cannot provide exact answers for our current question. That is, how can cognitive competencies required for goal-directed planning using visual attention and visual working memory develop through dynamic interactions among a set of different cognitive processes. Here, visual working memory has been known in neuroscience studies as active maintenance of visual information to serve the needs of ongoing cognitive tasks (Vogel & Machizawa, 2004; Fuster & Jervey, 1982). Our research group investigated this question in a previous study (Jung et al., 2019) and in the current study. The previous study investigated (1) how an arm robot with vision can learn a predictive model of the world by acting, using visual attention and working memory effectively, and (2) how goal-directed plans can be generated robustly using the acquired predictive model with generalization. The current study extended the previous study to address the question of how the robot can generalize in learning to deal with unlearned situations, such as manipulating unlearned objects. Let us consider previous findings first.

In the previous study (Jung et al., 2019), a network consisting of sub-module neural networks was assumed. More specifically, the whole network was comprised of a visual working memory (VWM), a visual attention module, an RNN module for predicting/generating various types of dynamically changing variables. Those variables include parameters for executive control, such as for visual attention control as well as visual masking control, and parameters related to visuo-motor pattern, including motor outputs, peripheral visual images as well as focused visual images in an attended area. Here, masking control of visual images means that each pixel value in a certain region is filtered with a specific parameter.

Each modular network is designed to be differentiable, and macroscopic connectivities among these modules are given.

Whole-network dynamics were modeled by following a framework of predictive coding (Mesulam, 1998; Rao & Ballard, 1999) and active inference (Friston et al., 2006) based on free-energy-minimization (FEP; Friston, 2005). Note that in predictive coding and active inference, attention is usually cast in terms of negentropy or precision of various likelihood probability distributions. Here, this is implemented in terms of selection or masking by effectively assigning zero precision to certain (non-attended) sources of sensory input.

This approach was taken because the FEP is considered one of the most influential theories that accounts for the underlying principle of cognitive brains using a generic Bayesian formula. Predictive coding accounts for perception of sensation in which perception is regarded as having been achieved when the error between sensory inputs and those re-generated by the generative model are minimized by inferring an optimal value of the latent state. On the other hand, active inference accounts for action generation wherein action on the environment minimizes the error between the desired sensation and the actual sensation. In this study (Jung et al., 2019) learning is conducted by following a predictive coding framework. More specifically, the whole network is trained to predict/reconstruct exemplar visuomotor sequences to minimize the reconstruction error by modifying connectivity weights of the whole network. This learning process also involves inferring optimal values of latent states of the whole network, which represent *intention* or *belief* for generating the exemplar sequences. Consequently, these latent variables also determine the temporal development of control parameters for visual attention, as well as visual masking. After learning converged, the active inference framework was used to generate goal-directed action plans to achieve given goal states. Optimal sequences of motor and control parameters for visual attention and visual masking are obtained by inferring latent variables of the whole network so as to minimize free energy while the connectivity weights are fixed.

In the current experiment, blocks of different colors were initially placed at random positions in the workspace, and an arm robot with a video camera was required to stack those blocks in an arbitrary configuration specified by the visual goal. Test trials for goal-directed planning were conducted with all connectivity weights of the network fixed after the robot was trained in various stacking tasks using the same blocks during tutoring by the experimenters. Separation of the training phase and the test phase was introduced for the sake of simplicity. Experimental results showed that the robot could achieve goal-directed action planning tasks successfully, showing a good generalization for novel situations. A particularly interesting finding was that whenever the robot grasped a block to move it, its visual attention went to the block autonomously while the static background image behind the block was saved in the VWM. This strategy emerged as a result of learning because it is beneficial for the network to allocate cognitive resources mainly for prediction of the visually focused area, i.e. an image of the block to be moved while the visual image of the remainder is saved in the VWM. This implies that the network may acquired a concept analogous to object permanency (Baillargeon et al., 1985) during the course of learning the exemplar.

However, this network cannot generalize well for certain situations, such as when novel blocks are introduced. More specifically, when blocks with unfamiliar colors are introduced, visuo-motor patterns of transferring such blocks from their grasped locations to a pre-selected location could not be generated in goal-directed planning, even though other features of the

blocks, such as size and shape were the same as those of the learned ones. Why did this happen?

This is because two mechanisms, learning to predict possible transformation of visual images associated with hand movement and memorizing contents of the transformation, are not dissociated. Therefore, the network is capable of imaging the visual transformation only for prior learned objects. In this situation, we added another VWM to support predictive generation of transformed images of given objects corresponding to their manipulation. We consider a new VWM wherein stored pixel patterns can be transformed for arbitrary rotation and translation by applying parameterised affine transformation. The parameter for transformation is provided from the predictive RNN module at every time step. By using such a VWM with the affine transformation mechanism, in order to generate desired transformations of visual images, the RNN module is just required to learn how images in the VWM can be manipulated, that is, it must learn to predict where the content at each pixel position in the current time step is mapped in the next time step, depending on the parameters provided to the affine transformation, but regardless of the content saved at each pixel in the VWM. Dissociation of learning about parameterised image transformations and memorizing image content should enhance generalization in image transformations, especially in cases of dealing with unlearned images, because the image transformations can be performed in a *content-agnostic* way, i.e., independently of image and content. Some cognitive neuroscience studies (Wilson et al., 1993; Ungerleider et al., 1998) have suggested that humans may use multiple VWMs separately, such as for preserving object images and for spatial or scenery images. Furthermore, from their neurophysiological experiments, Pailian et al. (2017) suggest that storage and manipulation are separable cognitive and neural mechanisms. These empirical studies may support the aforementioned modeling ideas at least partially.

The current study hypothesizes that addition of another VWM would improve generalization capability significantly by developing adequate information flows between the newly added VWM and other module networks, through learning. Particularly, we speculated that this newly added VWM might contribute to dynamic transformation of visual images of blocks, including novel ones, whereas the original VWM would store the static background image in the same way as in the original study. The current study evaluates this hypothesis by comparing performance in goal-directed action plan generation between cases with and without the second VWM and also by comparing dynamic mechanisms developed in these two cases.

The remainder of the manuscript is structured as follows. Section 2 introduces related studies and describes what novelties the current study inherits from them. Thereafter, we will present an overview and introduce details of our model in Section 3. Section 4 briefly describes dataset acquisition, followed by a presentation of experiments and their results. In Section 5, a summary of the current study is provided and limitations of the current model, as well as possible future studies, are discussed.

2. Related Work

As the previous section introduces briefly, the current proposed model uses predictive coding and active inference based on the free-energy-minimization principle, which incorpo-

rate a set of cognitive mechanisms, including goal-directed planning, visual working memory, and visual attention. The following explains these ideas by referring to related studies.

2.1. Free Energy Minimization

In the following, we appeal to many standard optimisation procedures, ranging from back-propagation of errors, through long short-term memory to variational RNNs. Although these schemes may appear bespoke and unconnected, they can all be understood as minimising variational free energy. Optimisation can be cast as a gradient descent on variational free energy (Isomura et al., 2020). Crucially, free energy gradients can, under simplifying (usually Gaussian) assumptions be cast as prediction errors. This means that minimising prediction errors destroys free energy gradients until a minimum is found. This general theme emerges in several forms ranging from predictive coding formulations of prediction error minimisation, to PID schemes for minimising proprioceptive error (Baltieri & Buckley, 2017), to back-propagation of errors in machine learning (Bengio & Fischer, 2015).

Negative free energy is known as an evidence lower bound in machine learning (Bishop, 2006). This means that minimising free energy is equivalent to maximising the evidence or marginal likelihood for a generative model of sensory data. The form of this model is our key focus here. In particular, its factorial structure produces a set of modules, each concerned with a particular domain of inference and learning. This characterisation corresponds to functional specialisation in the human brain. We leverage this by talking about working memory, attention, and other cognitive processes associated with sentient behaviour in humans.

Predictive Coding. Predictive coding presumes that perception can be achieved by minimizing possible discrepancies between top-down prediction and bottom-up sensory reality (Mesulam, 1998; Rao & Ballard, 1999). Predictive coding allows inference of hidden causes of sensation in the environment by comparison of sensory expectation and observed reality. Predictive coding rests on a hierarchical model in which prediction errors are propagated bottom-up through the hierarchy to optimize high-level representations that provide top-down predictions to guide successive predictions. It is assumed that the best explanation for sensory input is found when the top-down projection can explain as much of the bottom-up signal (at each hierarchical level) as possible (Brown et al., 2011).

Free Energy Principle. Based on the concept of Helmholtz and the view of the brain as a Bayesian inference machine, the free energy principle (FEP; Friston, 2005) introduces the concept of free energy as a tractable measure of the discrepancy between observed features of the world and representations of those features captured by generative models. More precisely, free energy exceeds than the model's negative log-evidence or *surprise* in sensory data, considering a model of how they were generated. The *evidence free energy* $\mathcal{F}$ for observed sensation can be written with decomposition into two terms as:

$$\mathcal{F} = -\underbrace{E_{q_{\boldsymbol{\varphi}}(z)}[\ln p_{\boldsymbol{\theta}}(X|z)]}_{\text{a) Accuracy}} + \underbrace{D_{\mathrm{KL}}[q_{\boldsymbol{\varphi}}(z)||p(z)]}_{\text{b) Complexity}}, \tag{1}$$

with hidden cause z, observation X and model parameters $\boldsymbol{\varphi}$ and $\boldsymbol{\theta}$. Here, it is essential to note that $p(z)/q_{\boldsymbol{\varphi}}(z)$ are an estimated prior/posterior probability distribution of the hidden

cause z before/after the observation of X. The accuracy term (a) includes a likelihood that relates sensory observations X to hidden causes Z and ensures that observations X have the same probability distribution as reconstructed by the approximate posterior $q_{\boldsymbol{\varphi}}(z)$ of latent variable z. The complexity term (b) facilitates regularization of the model by minimization of the divergence between the approximate posterior $q_{\boldsymbol{\varphi}}(z)$ and prior $p(z)$. The evidence free energy $\mathcal{F}$ can be minimized with respect to the posterior distribution $q_{\boldsymbol{\varphi}}(z)$ as:

$$q_{\boldsymbol{\varphi}}(z) = \text{argmin } \mathcal{F} \tag{2}$$

Friston (2005) argues that problems of perceptual inference, i.e., inferring the causes of sensory input, and perceptual learning, or learning a mapping of the cause to sensory inputs, can be resolved using exactly the same principle. Specifically, both inference and learning rest on minimizing the model's free energy.

However, predictive coding accounts only for perception, but not for action generation. In this regard, active inference developed recently by Friston and colleagues (Friston et al., 2006; Friston, 2010) proposes that action generation is a way to minimize prediction error by changing sensory inputs via adequately acting on the environment.

Active Inference. The *expected free energy* $\mathcal{G}$ is defined for the future as it considers possible effects of actions a applied to the environment. It can be represented with decomposition into two terms as:

$$\mathcal{G} = -\underbrace{E_{q_{\boldsymbol{\varphi}}(z)}[\ln p_{\boldsymbol{\theta}}(X(a)|z)]}_{\text{a) Accuracy in future}} + \underbrace{D_{\text{KL}}[q_{\boldsymbol{\varphi}}(z)||p(z)]}_{\text{b) Complexity}}, \tag{3}$$

where the first term represents the likelihood of experiencing a preferred sensation that is given extrinsically wherein sensation is a function of action a. The second term represents the same complexity term as the one in evidence free energy in Equation 1. In active inference, action a is optimized such that the expected free energy can be minimized as:

$$a = \text{argmin } \mathcal{G}. \tag{4}$$

Finally, by minimizing both the evidence free energy for the past according to Equation 2 and the expected free energy for the future according to Equation 4, perception and action generation can be carried out simultaneously by closing the loop between action and sensation (Baltieri & Buckley, 2017). For practical applications, like control of a robot, optimization of action at each time step to minimize the expected free energy by active inference can be facilitated by a lower-level controller, such as a PID controller. In this case, the PID controller receives the preferred proprioception of the next time step predicted by the network model, which is set as a target joint configuration of the robot at the next time step (Tani, 2003; Murata et al., 2017; Ohata & Tani, 2020). The PID controller computes necessary motor torques to minimize the error between the preferred sensation (i.e., the target joint configuration) and the actual one. This process is considered equivalent to Equation 4. By this means, active inference can account for reflex arcs which can be mechanized by a PID controller (Baltieri & Buckley, 2019).

Figure 1 outlines the aforementioned concept of closing the loop of action generation and perception through environments by showing the relationship between predictive coding

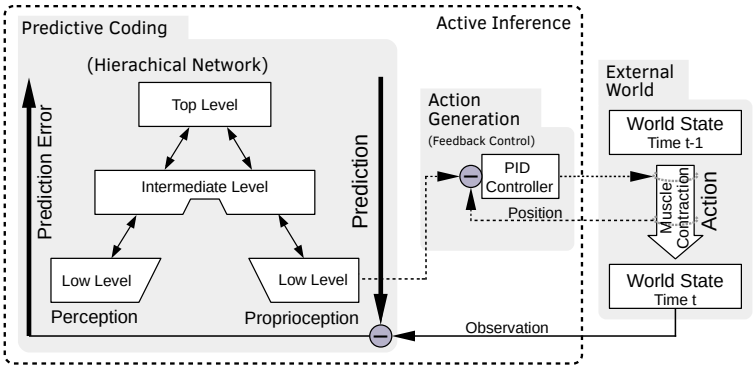


Figure 1: Illustration of closing action and perception by integrating predictive coding implemented in a hierarchical network and active inference in a PID controller demonstrated in (Tani, 2003; Murata et al., 2017; Ohata & Tani, 2020).

implemented by a hierarchical network model and active inference by a PID controller (Tani, 2003; Murata et al., 2017; Ohata & Tani, 2020). The hierarchical network predicts both exteroception and proprioception in the next time step, based on the current latent state. The PID controller receives the predicted proprioception as a target joint configuration and generates the corresponding movement of the robot. Accordingly, the environmental state changes and the hierarchical network senses the resultant exteroception and proprioception. Errors between the predicted and the observed sensation in both channels propagate from the lower level to the higher level, by which latent states in the network are updated toward minimizing the error.

Although the above-mentioned process of action generation and perception is based on only one-step-ahead prediction, the scheme can be extended to goal-directed planning to achieve a preferred state several steps ahead, as described in the following Section 2.2.

2.2. Goal-directed Planning using Active Inference

Early work on motor planning (Wolpert & Miall, 1996; Harris & Wolpert, 1998) proposes inference of optimal motor trajectories based on specific cost functions that include minimization of the discrepancy to a desired goal state added with regulation terms such as jerk minimization, position variance minimization against biological noise, motor torque minimization, and so on, using acquired forward models. Such models have been developed as inspired by neurobiological evidence (Ito, 1970; Miall et al., 1993; Wolpert et al., 1998).

However, combinatorial growth of complexity of the world poses challenges to scaling such models by employing hierarchical organization and multimodal sensory association with effective development of latent state trajectories (Finn & Levine, 2017; Nair et al., 2018; Jung et al., 2019).

Jung et al. (2019) recently developed a goal-directed planning scheme that is analogous to the framework of predictive coding and active inference. In their model, the hidden state of a recurrent neural network (RNN) at the initial step plays the role of the latent state that is assumed to have a Gaussian distribution with adaptive mean and standard deviation. Since this latent state, in terms of the initial state of the RNN, determines the succeeding visuo-proprioceptive sequence of all future time steps due to the initial sensitivity characteristics

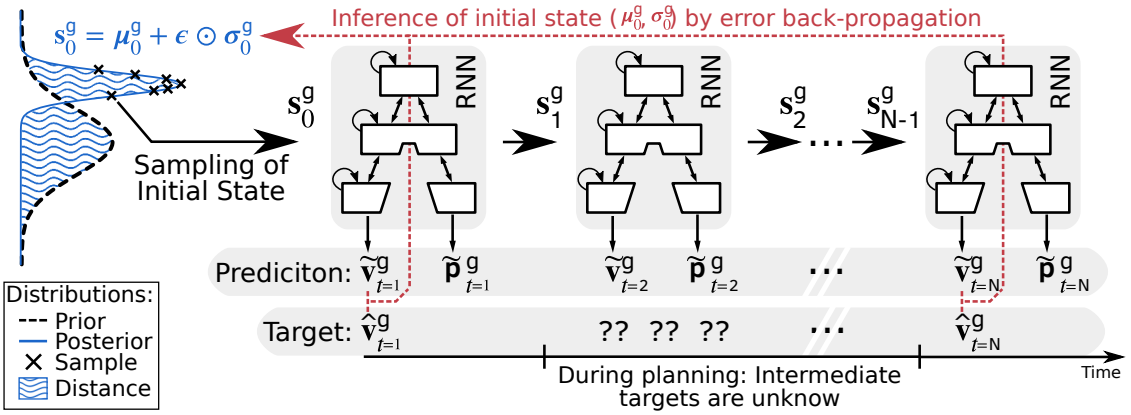


Figure 2: An illustration of goal-directed planning in (Jung et al., 2019). Sequences of proprioception $\tilde{\boldsymbol{p}}$, vision $\tilde{\boldsymbol{v}}$, and hidden states $\boldsymbol{s}$ that most likely attain a desired goal g are generated from the current posterior estimate (μ_0^g and σ_0^g) of the initial state $\boldsymbol{s}_0^g$. Back-propagation of the error between the desired visual outcomes $\hat{\boldsymbol{v}}$ and the generated one allows inference of an optimal posterior distribution of the latent state $\boldsymbol{s}_0^g$.

of an RNN as a deterministic dynamic system, this latent state can be interpreted as an intention or plan of the model to perform future actions.

For a given visuo-proprioceptive sequence, a posterior of the latent state to reconstruct the sequence can be inferred under the constraint of its prior as unit Gaussian. This is sometimes known as *planning as inference* (Kaplan & Friston, 2018). This idea of the posterior inference of the latent state using the initial hidden state of the RNN can be used in both learning and planning processes, as detailed below. It has been shown that this sort of probabilistic representation of the latent state is beneficial for gaining both generalization and robustness in learning as well as goal-directed planning (Jung et al., 2019). The following describes how training, planning, and action execution can be elaborated by following the framework of predictive coding and active inference mentioned previously in Jung et al. (2019).

In this study, training of the network by optimizing the connectivity weights is conducted first. More specifically, during training, the posterior of the latent state of the network $q_{\boldsymbol{\varphi}}(z)$ for each training sequence, as well as connectivity weights $\boldsymbol{\theta}$, are inferred to minimize the evidence-free energy shown in Equation 1. After training, while connectivity weights of the network are kept unchanged, tests to generate action plans to achieve given goals are conducted. Goals are specified in terms of visual pixel patterns that the robot can perceive by looking at the goal state of the block layout in the workspace. For this purpose the posterior latent state of the network is inferred to minimize the expected free energy shown in Equation 3. Figure 2 illustrates the mechanism. The posterior latent state $\boldsymbol{s}_0^g$ for a given goal g is inferred to minimize the error between the preferred visual state (i.e., the goal state) $\hat{\boldsymbol{v}}_{t=N}^g$ and the predicted state $\tilde{\boldsymbol{v}}_{t=N}^g$ at the distal step N and also the one between the visual state observed in the initial step $\hat{\boldsymbol{v}}_1^g$ and the predicted state $\tilde{\boldsymbol{v}}_1^g$ while the KL divergence between the posterior distribution $\boldsymbol{s}_0^g$ and the prior unit Gaussian distribution is minimized. Consequently, with respect to inference of the latent state of the network, given a goal to be achieved, the planning process inquires what visuo-proprioceptive sequence would most probably have been experienced.

Finally when the robot is activated in the workspace using the inferred plan, the PID controller generates adequate motor torques to minimize the discrepancy between proprioception predicted by the network during planning and the actual proprioception in terms of measured joint positions at each time step, as described previously. Hereafter, for simplicity, we use the term "motor sequence" to refer to an inferred or predicted proprioceptive sequence.

Although the RNN models used in the above-mentioned study are powerful in learning, generating, and inferring complex visuo-proprioceptive sequences, their capabilities are still limited, especially in dealing with high-dimensional visual image streams. Recently it has been suggested that adequate uses of visual attention and visual working memory could improve model performance significantly. Related studies that explore such possibilities are reviewed next.

2.3. Visual Working Memory and Attention

Recently, challenges of learning long-term dependencies with RNN models have motivated exploration of ways to incorporate working memory into RNN architectures. Methods that extended the idea of simple attention mechanisms with that of general memory structures for storage of more abstract representations are subsumed under the term *memory-augmented neural networks*, like Neural Turing Machine (NTM; Graves et al., 2014; Faradonbeh & Esfahani, 2019) and Differentiable Neural Computer (DNC; Graves et al., 2016), both of which are inspired by the *Von-Neumann* architecture. Further, recent work inspired by the concepts of the DNC addresses sequence-to-sequence translation and speech processing by allowing access to the network's memorized hidden states of all past time steps (Collier & Beel, 2019; Chien & Tsou, 2018; Le et al., 2018).

In the following, we review related studies that introduce visual working memory (VWM) and corresponding visual attention mechanisms. In comparison to previous approaches on visual long-term memory, which mainly operate in higher-level feature spaces or on the level of symbolic representations (e.g., Wersing et al., 2007), the Deep Recurrent Attentive Writer (*DRAW*) network (Gregor et al., 2015) introduces a VWM that is utilized as a sketch pad for writing, saving, and reading pixel images. The VWM is sequentially manipulated by an attention operation that focuses on a specific region of the visual sketch pad and an update of the attended visual information. The attention shifts at each time step are computed autonomously by means of mapping from the latent variable of the RNN wherein the mapping is developed during the training phase. As a result, attention shifts facilitate segmentation of complex patterns into a set of smaller sub-patterns and reuse of learned visual features at different spatial locations.

It has been shown that the *DRAW* architecture can be effective for generation of complex images with repetitive sub-patterns, e.g., generation of multi-digit number plates. Nevertheless, practical application of attention mechanisms and working memory have proven to be difficult, as attention mechanisms lead to instability and local minimum traps during training with the error back-propagation scheme (Tai et al., 2019; Finnveden et al., 2020).

Jung et al. (2019) investigated advantages of using VWM associated with RNNs for generating goal-directed action planning in a robotics task of vision-based object manipulation. In their model, possible visuo-proprioceptive sequences reaching desired visual goal images are generated based on predictive learning of exemplar sequences provided in the tutoring

phase. In generating a goal-directed visuo-proprioceptive sequence, RNNs associated with a VWM predict the visual image and proprioception of the next time step from the initial step to the distal time step.

More specifically, RNNs composed of stacks of LSTMs (Hochreiter & Schmidhuber, 1997) and convolutionary LSTM (Shi et al., 2015) predict a set of variables at the next time step, including proprioception, peripheral visual image, attended visual image and its attention parameters, and two types of pixel-wise masks. Each pixel-wise mask contains a weighting parameter at each pixel which is multiplied by RGB values at the pixel wherein the weighting parameter at each pixel is generated by the convolutionary LSTM. Prediction of the visual pattern of the next time step is computed by going through multiple paths using these predicted variables. First, the predicted peripheral visual pattern and the attended visual pattern are merged into a visual image panel using predicted attention parameters, including the position of the attention center in the pixel coordinate and the zooming ratio. Second, the content of the VWM is updated by interpolating the visual image in the panel and the currently preserved image in the VWM using one of the predicted pixel-wise masks for VWM. The ratio of preserving the current RGB values at each pixel in the VWM depends on the value of the predicted VWM pixel-wise mask for the pixel. The final prediction outputs of the visual image are generated by interpolating the visual pattern in the panel and that in the VWM using a further predicted pixel-wise mask. In this operation, the ratio of memory retrieval from the VWM at each pixel depends on the value of the predicted output pixel-wise mask for the pixel.

Introduction of the VWM system can greatly improve generalization and action planning performance of the whole system by effectively storing visual images occluded by movements of the robot in the VWM. Interestingly, the experimental results reveal that the VWM represents not continuous movement of the blocks, but sequences of a snapshot image of the block layout resulting from each block-stacking action. This represents succeeding sub-goals corresponding to the outcome of each block stacking action, as will be detailed later.

3. Proposed Model

Our proposed model design, as illustrated in Figure 3, is an extension and modification of Jung et al. (2019). In this design, we sought to introduce as few structural constraints as possible, in order to allow the system to develop necessary functions by itself in the course of end-to-end learning. The architectural design of the model elaborates especially on (1) a specific connectivity among different sub-modules; (2) newly considered parameterized attention mechanisms; and (3) fusion operations that allow it to merge visual predictions of RNNs with content of the VWMs.

The whole system consists of blocks of RNN-based generative models, an attention module, and two visual working memory modules. The RNN blocks consist of the associative LSTM (Hochreiter & Schmidhuber, 1997), a multi-layer LSTM, and multi-layer convolutionary LSTM cells (convLSTM; Shi et al., 2015). The associative LSTM (Figure 3(i)) located in the highest level of the whole network generates a sequence of top-down signals based on its initial latent state value (Figure 3(ii)) and sends it to both the multi-layer LSTM (Figure 3(iii)), and multi-layer convLSTM (Figure 3(iv)). The multi-layer LSTM predicts

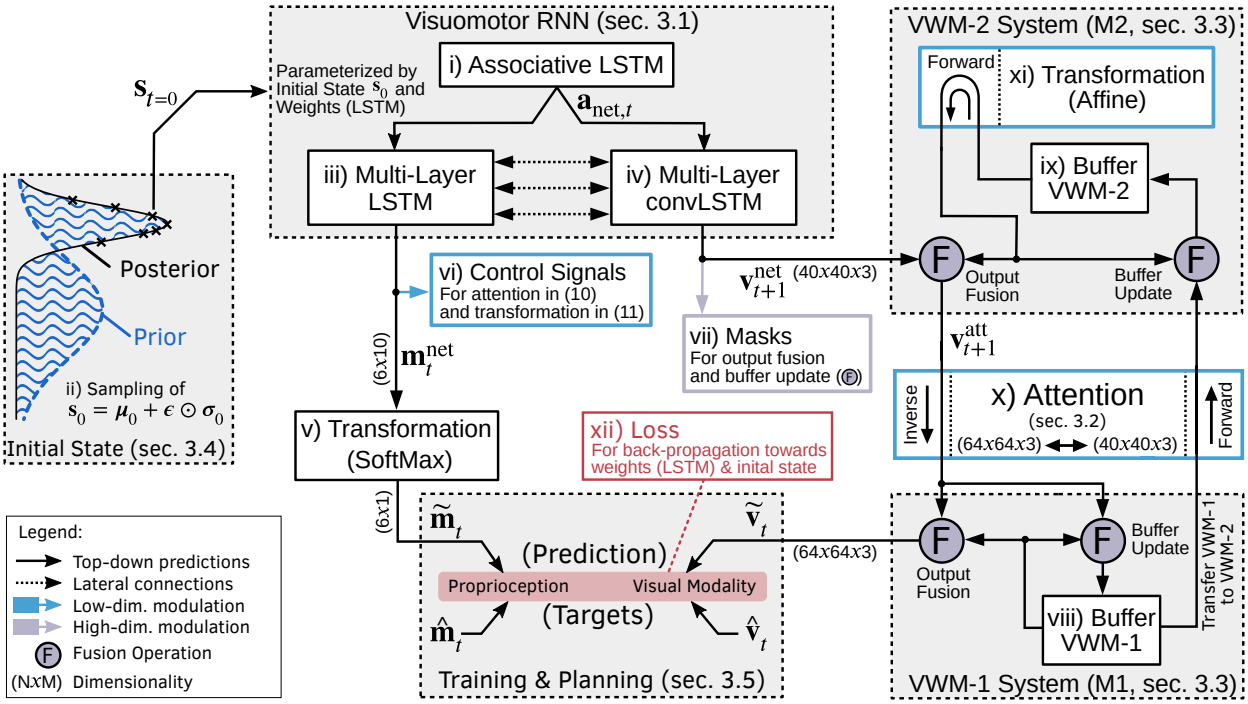


Figure 3: Schematic overview of the architecture of the proposed model, including the top-down and lateral pathways involved in generation of visuomotor predictions. The connectivity of the model facilitates development of two distinct mechanisms using visual working memory during training (7,8). Additionally generated low-dimensional parameterizations (blue) and pixel-wise masks (lilac), modulate the information flow of the system.

sequences of motor joint angles in terms of proprioception (i.e., motor sequence with simplicity) and multiple low-dimensional control signals. Proprioception is represented as sparse activation patterns of basis functions of a softmax encoding, as indicated in Figure 3(v). Predicted control signals modulate the information flow in the system by parameterization of visual attention and visual image transformation (Figure 3(vi)). Visual attention results in a dynamic adjustment of the pixel density of different regions in images that are generated by the RNN. As explained in the following (Section 3.2), visual attention allows the model to focus on and predict the visual appearance of manipulated objects in greater detail, while static parts of the generated images can be retrieved from the VWM. To overcome the restriction of representations in the VWMs to static content, we propose additional parameterized visual image transformations. The visual image transformation performs a pixel-wise transformation of images stored in a second VWM. As the result, the model becomes able to generate the visual image of object manipulation by means of adequate parameterization for the transformation applied to the visual image saved in this second VWM. For simplification, we restrict the model to affine image transformations, such as expected for our depicted pick-and-place scenario.

Predictions of the multi-layer LSTM are based on top-down signals received from the associative LSTM, lateral connections to the multi-layer convLSTM and its initial latent state values (Figure 3(ii)).

On the other hand, the multi-layer convLSTM predicts visual pixel images of the currently attended region and a set of masks (Figure 3(vii)). The masks are then used for mixing the predicted image by the convLSTM with those saved in each visual working memory. Again, this generation is based on the top-down signal received from the associative LSTM, lateral connections to the multi-layer LSTM as well as the initial latent state values (Figure 3(ii)).

By receiving the top-down prediction of visual image-related signals from the multi-layer convLSTM, two VWMs, VWM-1 (Figure 3(viii)) and VWM-2 (Figure 3(ix)), contribute to the final visual image prediction through their mutual interaction, as incorporated into a set of parameterized visual image operations, attention, inverse attention, fusion, and transformation. Attention (Figure 3(x)) is performed by application of the current attention filter, of which parameters are predicted by the multi-layer LSTM on the plain visual image for generating an attended image, and inverse attention is just an inverse transformation of it. Attention and inverse attention correspond to bottom-up and top-down projections, respectively, between neural representations of primary visual cues and of the abstracted visual representation in the convLSTM. Crucially, such bidirectional connections are required for inference of hidden states of the model within the predictive coding framework although its biological plausibility has not been identified yet.

The fusion operations (denoted by symbol Ⓕ) are to fuse two sources of visual streams with a pixel-wise mixing ratio represented by the corresponding mask generated from the multi-layer convLSTM. Fusion operations are utilized for the composition of the final prediction as well as for the update of the VWMs. A further affine image transformation (Figure 3(xi)) is applied to the visual image stored in VWM-2 wherein the transformation is parameterized by the prediction output of the multi-layer LSTM. Details of the top-down information flow of each block will be described later in Section 3.1.

In the learning process, updating the initial latent states and connectivity weights of the RNN blocks is performed with respect to minimization of the reconstruction error of the

visual and proprioceptive target sequence (Figure 3(xii)). To this end, back-propagation of the error (BP; Rumelhart et al., 1988) between the current prediction and the target is performed through the aforementioned top-down pathways inversely for updating values of the initial latent states. Connectivity weights of the whole network are optimized simultaneously. Thereby, the inference process consequently determines all parameters for the aforementioned operations of attention, inverse attention, fusion, and transformation at each time-step, since these parameters are generated by the RNN blocks as sensitive to their initial latent states as well. In goal-directed planning, the error in the form of a gap between the specified distal goal state and the mentally projected one is back-propagated through time (BPTT; Werbos, 1990). Back-propagation is performed for inferring the initial latent states in the RNN blocks (while the connectivity weights of the whole network are fixed) by which, plans, in terms of visuo-proprioceptive sequences for reaching the goal state, are generated. Inference mechanisms described here using bottom-up error signals for both end-to-end learning and targeted planning, become possible because the entire network is designed to be differentiable. It is expected that adequate cognitive processes for determining when and what to attend, as well as when and what to store or retrieve from the VWMs is fully developed through end-to-end learning during error minimization. In the following, details of each computational module, along with its connectivity with other modules, are described. Furthermore, procedures for training, as well as for planning and evaluating simulation experiments will be explained in detail.

3.1. Visuomotor Stream Prediction

The visuomotor stream network consists of visual (Figure 3(iv)) and proprioceptive pathways (Figure 3(iii)). Three layers of stacked LSTM (for proprioception) and convLSTM (for vision) modules are utilized. Each layer receives contextual information from neighboring layers, as listed in the following:

Top-down connectivity provides feedback from the next higher-level layer or from the associative layer of the model. Top-down computations propagate the prediction or belief of the network down to the sensorimotor level. A deconvolution operation is applied for expansion of the dimensionality of the neural activation of each layer to the increasing dimensionality of the next lower layer.

Lateral connectivity shares neural activation between visual and proprioceptive LSTM cells that are on the same layer of the model. Like calculations required for top-down processing, a deconvolution operation is applied to expand the lower-dimensional space of motor representations to fit the dimensionality of the feature-maps of the visual convLSTMs.

Bottom-up connectivity projects the neural activation of a lower layer of the model or the current sensory input (i.e. vision or proprioception) into the subsequent layer. The plain visual input image of the lowest layer of the model is transformed by the attention module, and projection into the next higher layer is performed by a convolution operation with a stride to reduce the sizes of feature maps and to facilitate spatiotemporal integration into higher layers.

Neural activation of the RNNs in the visual $\mathbf{v}^{\text{net}}_{l,t}$ and motor $\mathbf{m}^{\text{net}}_{l,t}$ pathways in the l^{th} layer at time step t are computed as described below.

The lowest layers receive visual input $\mathbf{v}_t$ and the softmax representation of the current joint

angle configuration $\mathbf{m}_t$:

$$\mathbf{v}^{\text{net}}_{l=0,t} = \text{ATT}(\mathbf{v}_t, \boldsymbol{\alpha}^{\text{att}}_t) \qquad \text{and} \tag{5}$$

$$\mathbf{m}^{\text{net}}_{l=0,t} = \text{SoftMax}(\mathbf{m}_t), \tag{6}$$

with visual attention transformation $\text{ATT}(\mathbf{v}_t, \boldsymbol{\alpha}^{\text{att}}_t)$ and its parameterization $\boldsymbol{\alpha}^{\text{att}}_t$, as defined in Section 3.2. The input of the lowest layer of the network, $\mathbf{v}_t$ and $\mathbf{m}_t$, depends on the execution mode of the network. It is either a one-step-ahead prediction $\widetilde{\mathbf{v}}_{t-1}$ and $\widetilde{\mathbf{m}}_{t-1}$ of the model, or the respective target $\widehat{\mathbf{v}}_t$ and $\widehat{\mathbf{m}}_t$ of the training data, as explained in more detail in Section 3.5 and Section 3.5.1. Neural activation in the visual pathway (convLSTM block) for layer $l = 1$ to $l = L$ is defined as:

$$\mathbf{v}^{net}_{l,t} = \begin{cases} \text{ConvLSTM}\left(\mathbf{v}^{\text{net}}_{l-1,t}, \mathbf{m}^{\text{net}}_{l,t-1}, \mathbf{a}^{\text{net}}_{t-1}\right), & \text{if } l = L \\ \text{ConvLSTM}\left(\mathbf{v}^{\text{net}}_{l-1,t}, \mathbf{m}^{\text{net}}_{l,t-1}, \mathbf{v}^{\text{net}}_{l+1,t-1}\right), & \text{otherwise.} \end{cases} \tag{7}$$

Neural activation in the proprioceptive pathway (LSTM block) is defined analogously as:

$$\mathbf{m}^{net}_{l,t} = \begin{cases} \text{LSTM}\left(\mathbf{m}^{\text{net}}_{l-1,t}, \mathbf{v}^{\text{net}}_{l,t-1}, \mathbf{a}^{\text{net}}_{t-1}\right), & \text{if } l = L \\ \text{LSTM}\left(\mathbf{m}^{\text{net}}_{l-1,t}, \mathbf{v}^{\text{net}}_{l,t-1}, \mathbf{m}^{\text{net}}_{l+1,t-1}\right), & \text{otherwise.} \end{cases} \tag{8}$$

In addition to an association of the visual and proprioceptive pathways by lateral connections in each layer of the RNN blocks, the model includes an associative LSTM for a combined representation of both pathways in the highest layer (Figure 3(i)). The associative LSTM is implemented as a standard LSTM and receives projections from the highest layers of the visual and proprioceptive RNN stacks. Its neural activation $\mathbf{a}^{\text{net}}_t$ is computed as follows:

$$\mathbf{a}^{\text{net}}_t = \text{LSTM}(\mathbf{v}^{\text{net}}_{l=L,t}, \mathbf{m}^{\text{net}}_{l=L,t}). \tag{9}$$

The output of the convLSTM block is the prediction of the attended visual image $\mathbf{v}^{\text{net}}_t$ and a set of masks, calculated as:

$$\mathbf{v}^{\text{net}}_t = \tanh(\text{Deconv}(\mathbf{v}^{\text{net}}_{l=1,t}), \tag{10}$$

$$\begin{bmatrix} \mathbf{g}^{\text{M1}}_t \\ \mathbf{g}^{\text{pred}}_t \end{bmatrix} = \text{ATT}^{-1}(\sigma(\text{Deconv}(\mathbf{v}^{\text{net}}_{l=1,t}), \boldsymbol{\alpha}^{\text{att}}_t) \quad \text{and} \tag{11}$$

$$\begin{bmatrix} \mathbf{g}^{\text{M2}}_t \\ \mathbf{g}^{\text{net}}_t \end{bmatrix} = \sigma(\text{Deconv}(\mathbf{v}^{\text{net}}_{l=1,t}), \tag{12}$$

with sigmoidal activation function σ. The masks $\mathbf{g}^{\text{M1}}_t$ and $\mathbf{g}^{\text{M2}}_t$ modulate the pixel-wise update of the VWMs. Further, the masks $\mathbf{g}^{\text{pred}}_t$ and $\mathbf{g}^{\text{net}}_t$ specify to what extent the final prediction of the plain visual image is based on the VWMs or the prediction $\mathbf{v}^{\text{net}}_t$ of the convLSTM block, as detailed in Section 3.3. The proprioceptive prediction $\mathbf{m}^{\text{net}}_t$ as well as the low-dimensional parameterizations $\boldsymbol{\alpha}^{\text{att}}_t$ and $\boldsymbol{\alpha}^{\text{M2}}_t$, which modulate the attention and transformation of VWM-2, are computed from the hidden states of the proprioceptive pathway as:

$$\mathbf{m}^{\text{net}}_t = \text{MLP}(\mathbf{m}^{\text{net}}_{l=1,t}), \tag{13}$$

$$\boldsymbol{\alpha}^{\text{att}}_t = \text{MLP}(\mathbf{m}^{\text{net}}_{l=1,t}) \quad \text{and} \tag{14}$$

$$\boldsymbol{\alpha}^{\text{M2}}_t = \text{MLP}(\mathbf{m}^{\text{net}}_{l=1,t}), \tag{15}$$

with MLP denoting a fully connected feed-forward network with one hidden layer of $N_{MLP} = 256$ nodes, layer normalization, and rectified linear activation functions. The final proprioceptive prediction is generated by a decoding of the softmax encoded predictions of the RNN:

$$\mathbf{m}_t = \text{SoftMax}^{-1}(\mathbf{m}_t^{\text{net}}). \tag{16}$$

3.2. Attention

Visual attention is performed by means of parameterization of scaling and focal position of the attention transformation. These parameters are generated by the multi-layer LSTM, which receives top-down signals from the associative LSTM located in the higher level, as described previously. Therefore, these parameters are actually determined by the initial states of these LSTMs in all levels through the top-down causality chain. This means that optimal parameters for visual attention during training and goal-directed planning are determined by means of the inference of optimal initial state values for the reconstruction error minimization. No explicit target values for the parameterization of the attention transformer are provided.

Jung et al. (2019) proposed distinct visual information processing with dorsal and ventral pathways, as described in Section 2.3. The dorsal stream processes a down-scaled, low-resolution, peripheral visual input image, whereas the ventral stream utilizes a spatial transformer network to process only an attended region of the visual input image, by cropping and zooming. We presume that although this idea of two visual pathways is biologically plausible (two-streams hypothesis; Goodale & Milner, 1992), implementation of this concept in a synthetic model may not be always necessary, depending on the given tasks. Our preliminary studies showed that visual image transformations by attention and inverse attention are among the most important elements for successful development of visual working memory function during end-to-end learning (a comparison is presented in Appendix F). In the current study we propose a modified spatial transformer network (STN; Jaderberg et al., 2015) that performs a non-linear transformation of the input image, such that a specific region of the image can be focused with a high pixel density, while the unfocused regions of the image can be represented with a lower pixel density as well.

Following the preliminary study, which showed that the novel attention scheme using the modified transfer network provides a better performance, we employed a spatial transformer network $\text{ATT} : U \in \mathcal{R}^{N_{in} x N_{in}} \mapsto V \in \mathcal{R}^{N_{net} x N_{net}}$ for generating a composite representation of the peripheral and focal visual image, each with a different pixel density ratio. The pixel-wise transformation of a visual image from input location U to output location V is defined by a modified grid generator (Jaderberg et al., 2015), parameterized by $\boldsymbol{\alpha}_t^{\text{att}}$. Further implementation-specific details are listed in Appendix A.

3.3. Network-wise processing of vision

The multi-layer convLSTM outputs predictions of attended visual images along with a set of masks used for fusion of the visual images. The final prediction of the plain visual image in the next time-step is generated by performing further network-wise operations on this vision-related information, including forward and inverse attention shifts, affine transformation, fusion, and buffering using two visual memory buffers, one in the unattended (VWM-1) and

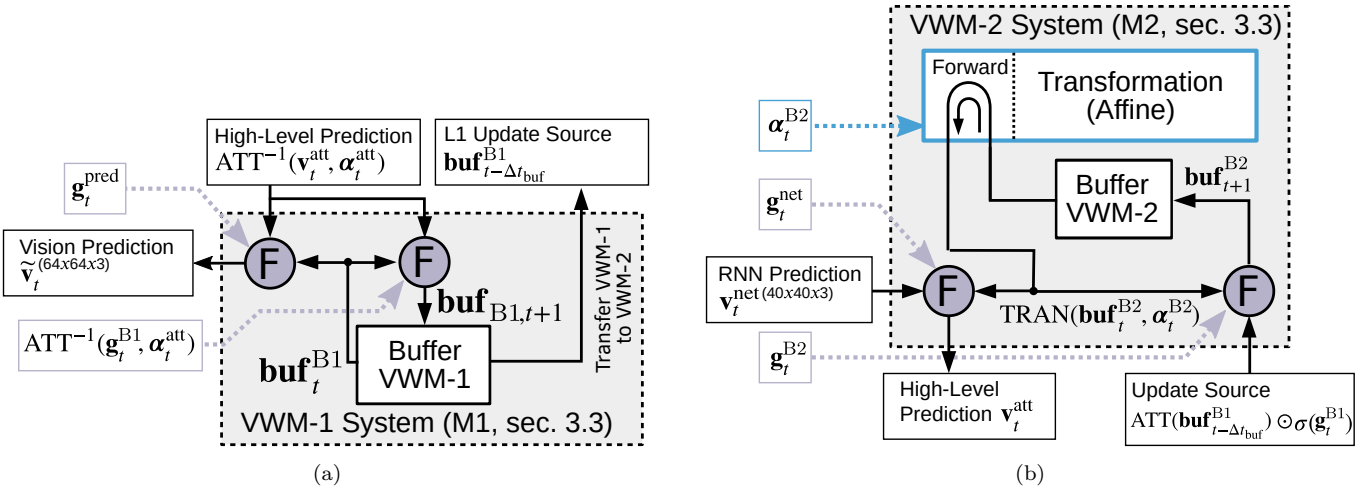


Figure 4: Detailed view of network connectivity related to visual working memory VWM-1(a) and visual working memory integration of VWM-2 (b). This figure depicts detailed views of the system diagram in Figure 3.

the other in the attended (VWM-2) visual feature space of the model. Details of network-wise operations can be described by the following equations:

$$\begin{aligned}\mathbf{vwm}^{\text{M1}}_{t+1} = (1 - \text{ATT}^{-1}(\mathbf{g}_t^{\text{M1}}, \boldsymbol{\alpha}_t^{\text{att}})) \odot \mathbf{vwm}_t^{\text{M1}} \\ + \text{ATT}^{-1}(\mathbf{g}_t^{\text{M1}} \odot \mathbf{v}_t^{\text{att}}, \boldsymbol{\alpha}_t^{\text{att}}).\end{aligned} \tag{17}$$

Equation 17 describes how the contents of VWM-1, $\mathbf{vwm}^{\text{M1}}_{t+1}$, can be updated, where $\mathbf{g}_t^{\text{M1}}$ denotes a pixel-wise mask and ATT^{-1} performs inverse attention with arguments of the predicted attended visual image $\mathbf{v}_t^{\text{att}}$ and attention parameter $\boldsymbol{\alpha}_t^{\text{att}}$. Masking of the visual streams is performed by the element-wise multiplication operator denoted by symbol $\odot$. A visualization of the network connectivity related to VWM-1 is depicted in Figure 4a.

$$\begin{aligned}\mathbf{vwm}^{\text{M2}}_{t+1} =& \mathbf{g}_t^{\text{M2}} \odot \text{TRAN}(\mathbf{vwm}_t^{\text{M2}}, \boldsymbol{\alpha}_t^{\text{M2}}) \\ &+ (1 - \mathbf{g}_t^{\text{M2}}) \odot \text{ATT}(\mathbf{vwm}^{\text{M1}}_{t-\Delta t_{\text{vwm}}}, \boldsymbol{\alpha}_t^{\text{att}}) \odot \sigma\left(\mathbf{g}_t^{\text{M1}}\right).\end{aligned} \tag{18}$$

Equation 18 describes how VWM-2, $\mathbf{vwm}^{\text{M2}}$, can be updated, as outlined in Figure 4b. The variable $\mathbf{g}_t^{\text{M2}}$ denotes a pixel-wise mask that defines the fusion of transformed contents $\text{TRAN}(\mathbf{vwm}_t^{\text{M2}}, \boldsymbol{\alpha}_t^{\text{M2}})$ of VWM-2 with time delayed $(t - \Delta t_{\text{vwm}})$ contents of VWM-1, denoted as $\text{ATT}(\mathbf{vwm}^{\text{M1}}_{t-\Delta t_{\text{vwm}}}, \boldsymbol{\alpha}_t^{\text{att}}) \odot \sigma\left(\mathbf{g}_t^{\text{M1}}\right)$. The update of $\mathbf{vwm}^{\text{M2}}$ is restricted to recently modified contents of VWM-1 by masking of $\mathbf{vwm}^{\text{M1}}_{t-\Delta t_{\text{vwm}}}$ with $\sigma\left(\mathbf{g}_t^{\text{M1}}\right)$.

Notes on the biological plausibility of the implementation of VWMs: For the proposed computational model, we refer to a simplified implementation of the VWMs as plain buffers. Abstraction of underlying neurological details of memory formation is a common design choice to reduce computational efforts in cognitive modelling of complex learning systems, e.g. (Hochreiter & Schmidhuber, 1997; Gregor et al., 2015; Jung et al., 2019). Further, underlying neurological principles of memory formation and maintenance are still very much under discussion and unknown to large extent. But we hope that future empirical neuroscience

studies can evaluate predictions made from studies in cognitive modeling. Maintenance, update, and read-out operations of the content of the VWMs are based on primitive network operations such as gating, time-delays, normalization, and non-negative fusion of pathways. Further, the fusion operations, as required for the memory read-out, relate in their functionality to the competitive-layer model (Wersing et al., 1997, CLM), as they perform pixel-wise winner-take-all operations between two sources of input features. As a result of our experiments, segmentation of different objects in the visual image similar to segmentation by the CLM can be observed, as discussed in Section 4.3.3.

The transformation TRAN($\mathbf{vwm}_t^{\mathrm{M2}}, \boldsymbol{\alpha}_t^{\mathrm{M2}}$) performs an affine projection of the input image $\mathbf{vwm}_t^{\mathrm{M2}}$ by applying a spatial transformer network (STN; Jaderberg et al., 2015) with parameterization $\boldsymbol{\alpha}_t^{\mathrm{M2}} \in \mathcal{R}^4$ covering independent scaling and shift factors for both image dimensions.

Our preliminary studies showed that the introduced delay Δt_{vwm} is crucial for development of content transfer from VWM-1 to VWM-2 during training. A short delay Δt_{vwm} overcomes the transition phase between approaching an object (predictions of the object's appearance are based on VWM-1) and moving the object (predictions are based on the RNN and VWM-2) in which representations of manipulated objects in VWM-1 already start to fade, as discussed in more detail in Section 4.3. If not otherwise noted, we refer to a delay of $\Delta t_{\mathrm{vwm}} = 5$ in our work.

Prediction $\mathbf{v}_t^{\mathrm{net}}$ of visual images in the attended visual feature space is performed by a fusion of the predictions made by the convLSTM and the contents of VWM-2, defined as follows:

$$\begin{aligned}\mathbf{v}_t^{\mathrm{att}} = &\mathbf{g}_t^{\mathrm{net}} \odot \mathbf{v}_t^{\mathrm{net}} + \\ &(1 - \mathbf{g}_t^{\mathrm{net}}) \odot \mathrm{TRAN}(\mathbf{vwm}_t^{\mathrm{M2}}, \boldsymbol{\alpha}_t^{\mathrm{M2}}).\end{aligned} \tag{19}$$

By applying the inverse of the attention transformation ATT^{-1} to the predicted attended image $\mathbf{v}_t^{\mathrm{att}}$, fusion with this transformed image and the image saved in VWM-1 becomes possible, generating the final prediction output for the plain visual image at the next time step $\widetilde{\mathbf{v}}_t$, computed as:

$$\begin{aligned}\widetilde{\mathbf{v}}_t = &\mathbf{g}_t^{pred} \odot \mathrm{ATT}^{-1}(\mathbf{v}_t^{\mathrm{att}}, \boldsymbol{\alpha}_t^{\mathrm{att}}) + \\ &(1 - \mathbf{g}_t^{\mathrm{pred}}) \odot \mathbf{vwm}_t^{\mathrm{M1}}.\end{aligned} \tag{20}$$

3.4. Inference and Sampling of Initial Sates

As discussed in Section 2, we utilize the initial state sensitivity characteristic of dynamic systems for sequence generation in RNNs in order to represent task variability. This means that variation in the initial states of RNNs accounts for variation in sequences generated by the RNNs. The proposed model is trained with a set of successful visuomotor sequences, i.e. sequences that achieve a given goal. Training involves inference of connectivity weights as well as initial states of the network. Connectivity of the network is assumed to be fixed after training and same assumption applies to generation of all training sequences. Two types of initial states are inferred during training, as outlined in Figure 3(ii): 1. a common prior initial state that represents the distribution of all training sequences, and 2. each different posterior initial state for representation of each training sequence. After successful training, the

model is capable of regenerating all training sequences by setting the initial states with those posteriors inferred in the training phase. In case of planning for novel goals, corresponding initial states need to be inferred for generation of corresponding visuo-proprioceptive (i.e. visuomotor) sequences for execution of goal-directed actions. Preparation of motor plans by inferring appropriate initial states can be seen as analogous to motor planning in the brain, as discussed in Shima et al. (2007).

For implementation, we refer to a variational Bayes approach for probabilistic representations of latent states, as formulated for the variational autoencoder (VAE; Kingma & Welling, 2014) and its extension to continuous RNN models (Murata et al., 2013, 2017). The initial state $\mathbf{s}_0^g$ is encoded as a probability distribution and is sampled using the reparameterization trick (Kingma & Welling, 2014) to allow back-propagation of reconstruction errors:

$$\mathbf{s}_0^g = \boldsymbol{\mu}_0^g + \boldsymbol{\epsilon} \odot \boldsymbol{\sigma}_0^g \quad \text{with} \tag{21}$$

$$\epsilon \sim \mathcal{N}(0, \boldsymbol{I}). \tag{22}$$

The initial state $\mathbf{s}_0^g$ for generation of optimal goal-directed actions for goal g is defined by its mean $\boldsymbol{\mu}_0^g$ and standard deviation $\boldsymbol{\sigma}_0^g$, with auxiliary noise $\boldsymbol{\epsilon}$ sampled from a normal distribution. The probabilistic representation of the initial latent state allows computation of a belief or an estimate of precision in learning, as well as generation of each sequence. This can provide significantly greater robustness in behavior of the model. Such benefits can be observed especially in dealing with noisy or stochastic situations (Murata et al., 2013) and in comparison to previous models that refer to deterministic representations of the initial latent state (e.g. Arie et al., 2009; Choi et al., 2018).

3.5. Training

The proposed model is trained in order to generate multiple visuomotor sequences in relation to their corresponding initial states. During the training process, the prior initial state common to all training sequences, and the posterior initial states for each of them are inferred. Each initial state is parameterized with a mean and a standard deviation, which are updated during the training procedure, i.e. the prior and the posterior initial states are updated simultaneously as in Denton & Fergus (2018). The weights, biases, and initial states of the model are updated to minimize the evidence free energy, as discussed in Section 2.2, including the visuomotor reconstruction error as well as the Kullback-Leibler divergence between the prior and the posterior initial states.

Our model is fully differentiable and can be trained by state-of-the-art gradient descent methods, like the ADAM optimization technique (Kingma & Welling, 2014).

The loss L^g for the g-th goal of the training data is calculated as:

$$L^g = L^g_{\mathbf{v}} + L^g_{\mathbf{m}} + \beta D_{\mathrm{KL}}(q_{\phi^g}(\mathbf{s}_0^g)||p_{\boldsymbol{\theta}}(\mathbf{s}_0)) + L^n_{|\cdot|}, \tag{23}$$

with parameterization of the posterior and prior initial state, ϕ^g and θ, respectively. The function $D_{\mathrm{KL}}(\cdot)$ denotes the Kullback-Leibler divergence and the hyper parameter β adjusts the balance between the minimization of the prediction error and the divergence between prior and posterior, as previously proposed in Denton & Fergus (2018). Note the formal

similarity between Equation 23 and Equation 1 (and 3) where the accuracy corresponds to the various components of negative loss, L^g.

The functions of the visual $L^g_{\mathbf{v}}$ and proprioceptive $L^g_{\mathbf{m}}$ reconstruction loss are defined as follows:

$$L^g_{\mathbf{v}} = \sum_{t=1}^{T^g} L^g_{\mathbf{v},t} = \sum_{t=1}^{T^g} \mathbf{s}^{\text{att}}_t \odot (\widehat{\mathbf{v}}^g_{t+1} - \widetilde{\mathbf{v}}^g_t)^2 \quad \text{and} \tag{24}$$

$$L^g_{\mathbf{m}} = \sum_{t=1}^{T^g} L^g_{\mathbf{m},t} = \sum_{t=1}^{T^g} D_{\text{KL}}(\text{SoftMax}(\widehat{\mathbf{m}}^g_{t+1}) || \widetilde{\mathbf{m}}_{\text{net},\mathbf{tg}}). \tag{25}$$

Lengths of trajectories are denoted by T^g, the proprioceptive targets for time step t in softmax encoding by $\widehat{\mathbf{m}}^g_{\text{net},t}$, and visual targets by $\widehat{\mathbf{v}}^g_t$. Additionally, we introduce $\mathbf{s}^{\text{att}}_t$ to balance contributions of the focal and peripheral regions of the visual error signal to impede an over-representation of back-propagated gradients of the focal area, caused by the attention transformation, as detailed in Appendix B. The regularization term of the loss function includes the ℓ_1-Norm of the masking operator $\mathbf{g}_{\text{net},t}$ to prefer predictions based on VWM-2 over predictions of the RNN blocks, even though pixel-wise predictions from RNN blocks may achieve a smaller prediction error on the training data. The regularization loss is defined as $L^g_{|\cdot|} = {}^1\!/\!_T \sum_{t=1}^{T^g} (\lambda_{|\cdot|,t} |\mathbf{g}_{\text{net},t}|_1)$. The magnitude of the regularization factor $\lambda_{|\cdot|,t}$ is calculated by application of a sigmoidally shaped function on the current training epoch. A low regularization $\lambda_{|\cdot|,t}$ at the onset of learning supports development of VWM-1 and an increasing and bounded regularization factor toward the final learning phase results in a preference of contributions of VWM-2 over predictions from RNN blocks, if applicable. If not otherwise noted, we apply ℓ_1 regularization, and scaling factor $\lambda_{|\cdot|,e}$ is defined in relation to the current epoch e as:

$$\lambda_{|\cdot|,e} = \begin{cases} 0, & \text{if } e < 3750 \\ 4.0 \cdot min(1, {}^{(e\,-\,3750)}\!/\!_{1250}), & \text{otherwise.} \end{cases} \tag{26}$$

The proposed loss L^g is defined for a single training sample, but can be trivially extended for a mini-batch learning configuration. An overview of implementation of the training process is depicted in Algorithm 1 in Appendix C. The training phase follows the closed-loop training scheme (Yamashita & Jun, 2008) to minimize the prediction error and thereby to improve mental simulation capabilities of the model. For closed-loop training, predictions of the model are fed back to the model as inputs for the next time step. However, as a sole closed-loop optimization of the model can lead to a strong divergence of network states (instability in training in particular is critical for early phases of the training process), a mixture of model predictions and training targets is used as a feedback signal for the next time step. Therefore, the feedback signal of the network is calculated as follows:

$$\mathbf{v}^{l=0}_t = 0.9\widetilde{\mathbf{v}}_{t-1} + 0.1\widehat{\mathbf{v}}_t \quad \text{and} \tag{27}$$

$$\mathbf{m}^{l=0}_t = 0.9\widetilde{\mathbf{m}}_{t-1} + 0.1\widehat{\mathbf{m}}_t. \tag{28}$$

3.5.1. Planning

As previously discussed (Section 2.2), planning for an action, given a novel goal is conducted by minimizing the expected free energy and searching an optimal posterior of the

initial state of the model. In our work, the goal is specified in terms of desired visual sensation at the end of the predicted sequence.

The optimal posterior is found if a few steps of the initial visuomotor sequence as well as the visual sensation at the goal step, both generated by the network, match those specified for each task trial with minimal error while the KL divergence between the posterior and the prior can be minimized as well.

In the beginning of the iterative search of the posterior initial state, its value is initialized to the prior estimate as inferred during training through Equation 23.

Consequently, the loss function for plan generation is defined as:

$$L_p^g = \sum_{t=1}^{T_g} (L_{\mathbf{v},t}^g + L_{\mathbf{m},t}^g) + \beta D_{\mathrm{KL}}(q_{\phi^g}(s_0^g)||p_{\boldsymbol{\theta}}(s_0)) + L_{v,T_e}^g, \tag{29}$$

with parameters for the initial state denoted as ϕ^g for the g-th goal of the test cases. The length of the initial sequence is denoted as T_g and T_e specifies the time step in which the desired visual image for the goal configuration should appear. Further details for implementation of the planning process are depicted in Algorithm 2 in Appendix C. Note that in this case, the function of the generative model is fixed by fixing its weights. Then, updates of the initial state of the model in order to minimize the loss L_p^g result in trajectories that reach a specified goal, as they minimize the visual discrepancy between a desired goal state and the predicted state at the final time step.

4. Experiments

The current experiments introduce a block-stacking scenario in which the task is to arrange three blocks in a tower configuration. Successful stacking operations of a robotic actuator have been recorded in a real-world setup, as presented in Figure D.9. During training, only three blocks of different colors (red, green and blue) are introduced and the model evaluation assesses the generalization to new block positions and stacking orders. In addition to the recorded dataset that includes three block colors, we prepare an additional augmented version of the dataset in which colors of the objects are replaced by one randomly permuted in the color plane (see the details in Figure D.10). The augmented data set allows an evaluation of the generalization capability of the models in dealing with objects having unfamiliar appearances, such as a new color. A discussion on further conditions, including experiments that explore generalization to more complex visual appearances such as textures, is shown later (Appendix Appendix G).

In the current study, two types of experimental evaluations are conducted. First, a descriptive evaluation of system performance is performed. The purpose of the descriptive evaluation is to explore possible developments of cognitive mechanisms in the network, including a scheme developed to manipulate visual images using visual working memory. Second, a quantitative evaluation is conducted wherein comparative analysis of performance in goal-directed planning is carried out under various conditions. Performance is measured by computing errors generated between the ground truth and the visuomotor sequences inferred for a set of goal-directed planning cases. Note that the terms "motor sequences" and "proprioception" are used interchangeably in our work, as the inferred proprioceptive sequences

are used as targets for the feedback controller of the robot's actuators and low-level control is neglected, as previously disused in Section 2.1. In order to evaluate the generalization capability of the trained network, which is required especially for content-agnostic information processing, we conduct goal-directed action planning experiments under novel task configurations by introducing objects with novel colors, as described previously. These evaluations are conducted with comparisons among three different models, including the current model, the previous model with one visual working memory, as proposed by Jung et al. (2019), and one without any visual working memory. It is expected that the currently proposed model using two visual working memory modules should show a significantly better generalization performance compared to those with only a single memory module for the following reasons: first, neuroscience literature suggest the human uses multiple specialized VWMs rather a single one (Wilson et al., 1993; Ungerleider et al., 1998); and second, the implementation of a second VWM and its transformation in the attended visual feature space of the model allows a dissociation of learning about the parameterized image transformations and memorizing the image content, as discussed in Section 1.3.

4.1. Data set acquisition and experimental setup

For evaluation, the system is confronted with a complex multimodal sensorimotor task. As the task design in Figure D.9 shows, a robotic actuator (Torobo Arm; Tokyo Robotics Inc., 2020) is mounted in front of a table, on which three box objects are placed at random positions. The actuator is commanded in joint-space position control in order to perform two successive stacking operations of the randomly placed objects, each of which results in a tower of those objects. For these experiments, we utilize 6 of the 7 degrees of freedom of the robot, since an additional rotation of the end-effector is not required for object manipulations of the task.

Test and training trajectories for pick-and-place manipulation for the block stacking task are generated based on kinesthetic teaching of the robot. Recording is performed at 20Hz and downsampled 7 times to reduce computational and memory costs. Automated generation of trajectories results in variation of ≈10% of their lengths. Please note that the proposed model is based on a RNN and allows representation of sequences with variable lengths. The final preprocessing, as detailed before, results in visuomotor sequences with a length of $T_e = 100 \pm 5$ steps for training and test evaluation of the models. Permutation of the stacking order of three colored blocks (red (r), green (g) and blue (b)) of size $5cm^3$ resulted in six possible tower configurations. Training is performed for four tower configurations (RGB, RBG, GBR, GRB) and excluded the configurations (BGR and BRG), which are only included in the test data set. Block positions are sampled from 10x10 and 8x8 grids for training and testing, respectively, in order to test generalization for previously unseen spatial location distributions of objects.

In addition to joint trajectories, the visual frame sequence of an external camera that shows the objects and the robot actuator interacting with them are stored in a dataset. The recorded dataset contains 300 task configurations with randomly placed objects and randomly selected tower configuration as goals for training, and 45 task configurations used for evaluation that are distinct from the training set. The augmented test data set is generated by a random selection of one of 5 permutations of the color planes of visual sequences for all

45 sequences of the test data set. During training, temporal sequences of successful sensorimotor signals that fulfill the task goals are presented. The trained network was evaluated for its ability to generate plans using the schemes described in Section 3.5.1 to achieve novel goal configurations, starting from novel object arrangements, including cases using novel object colors. Quantitative evaluation was made based on the measured error between the generated plan of action and the ground truth.

4.2. Implementation details

Network parameterization: The associative LSTM at the top of the RNN module, which integrates the visual and proprioceptive streams, contains a single LSTM layer with 512 neurons. The proprioceptive and visual pathways of the recurrent neural network are based on three layers of multi-layer LSTM cells and multi-layer convLSTM cells, respectively. The multi-layer convLSTM includes 16, 32, and 64 feature maps from the sensory to the highest layer. To project features to the next higher layer, convolutional kernel size, stride, and padding sizes were set to 5x5, 2x2, and 2x2, respectively. Deconvolutional kernel size stride, and padding to project features to the next lower layer were set to 6x6, 2x2, and 2x2, respectively. For lateral connections from hidden states of the motor pathway to the visual pathway and the projection from the associative LSTM, convolutional kernel sizes are selected in such way as to match the feature dimensionality of the respective layer of the visual pathway. Forward transformation of the attention transformer downscales the resolution by a factor of 0.75 (32x32 to 24x24 pixels), or 0.625 (64x64 to 40x40 pixels), to fit the size of prerecorded data sets. The proprioceptive pathway is based on a multi-layer LSTM with 512, 256, and 128 neurons from the lowest to the highest layer. For prediction of proprioception and parameterization of the attention modules, a multilayer perceptron (MLP) with one hidden layer of 256 neurons, layer normalization (LN; Ba et al., 2016) and rectified linear unit (ReLU) activation functions is utilized. Note: We observed that an under-representation (low dimensionality of convolutional hidden layers) results in a colorless representation of the scene, but we have not systematically analyzed this effect.

Training of the network: Training of the model by minimizing the loss function of Equation 23 was performed using the ADAM optimizer (Kingma & Welling, 2014). Optimization of initial states, weights, and biases was performed for over 4500 epochs, until convergence of learning. The learning rate was set to 5×10^{-4} and the hyper-parameter β was set to 1×10^{-5}. To prevent instability during training (i.e., *exploding gradient problem*) we performed gradient clipping (Pascanu et al., 2013), which re-scales gradients based on the ℓ_2-norm in case the norm of the gradients exceeds 0.2. The mean and standard deviation of the prior and posterior initial states were set to 0 and 1, respectively.

Planning and Evaluation for Unseen Situations: Planning of actions for a previously unseen goal is conducted using the loss L_p, as defined by Equation 29. The posterior initial state of the model is optimized for the best match of the visual images of the first $T_g = 5$ time-steps (the state of the world before a goal-directed action was executed) and a desired visual goal image at the end of the sequence at the final time-step T_e. An initial tentative value of the posterior is set with the value of the prior that was acquired as a common value for all sequences of the training set and represents the distribution of all training sequences, as described in Section 3.5. The successive update of the posterior performed for the planning process is explained in detail in Section 3.5.1. In order to update the initial state estimate at

each epoch, the visuomotor sequence is generated 16 times by repeated stochastic sampling of the posterior initial state, the mean and variance of which were inferred. One of the sampled initial states that results in the lowest planning error after 50 epochs of inference is selected as the final result of the planning process for the given goal. The visuomotor sequence generated from the final initial state obtained is considered the final visuomotor sequence plan. Note that even though the loss function L_p inhibits deviations only in the visual modality for the final configuration, reasonable generation of motor/proprioceptive sequences can be expected since training was performed using an association of vision and proprioceptive sequences.

4.3. Descriptive Evaluation of Results

In this subsection, we first discuss a qualitative analysis of the neural mechanisms self-organized in the model after successful training. Thereafter, we will show the results of a comparison to previous models and discuss the importance of VWM-2, the second visual working memory that is introduced in our study with the aim to improve the generalization capabilities of the model.

4.3.1. Self-organized neural mechanisms

To assess the properties of our proposed system, we analyze the internal states of the visual pathway and the output of the model during execution of inferred plans for previously unseen tasks. Due to the task nature, involving two consecutive stacking actions, the objects switch their ongoing roles from being part of the background to being manipulated objects in the visuo-proprioceptive sequences, generated as plans.

One exemplary evaluation is shown in Figure 5, which displays the internal states of the visual pathway, the prediction of the proprioceptive pathway, and the error between the generated plan and the ground truth. The evaluation visualizes the mental simulation of the generated plan and the respective expected visual perception for a previously unseen arrangement of objects and unknown colors, e.g., the newly introduced orange block. Figure 5(i) allows a comparison of the ground truth joint-angle trajectories of the augmented test data set with the inferred trajectories of the planning process that is shown in Figure 5(ii). Trajectories #0-4 (blue, orange, green, red, purple) represent joint angles of all 5 active rotary joints of the robot arm and joint angle #5 (brown color) refers to the one of the linear actuators of the robot gripper. A visualization of the mismatch between the ground truth of the visual stream of the augmented test data set (Figure 5(iv)) and the inferred visual perception during plan execution (Figure 5(v)) is shown in Figure 5(iii). The visual stream shows every 8th time-step of the generated sequence of the model. Figure 5(v) marks the current focal area, in terms of size and position of the attention transformation, indicated by a red square. Parameterization of the attention transformer is generated as an additional output of the multi-layer LSTM, as outlined in Figure 3(5).

Figure 5(vi) shows the content of the visual buffer of VWM-1. Figure 5(viii, ix and x) show dynamics of the visual buffer of VWM-2, including the update and readout pathways. At each time-step, the visual buffer of VWM-2 (Figure 5(ix)) can be updated with a transformation of its own content or the time-delayed and masked content of VWM-1, as indicated by Figure 5(vii.i). The readout of the visual buffer of VWM-2 developed through the self-organized masking operation is shown in Figure 5(viii). Figure 5(x.i and x.ii) detail

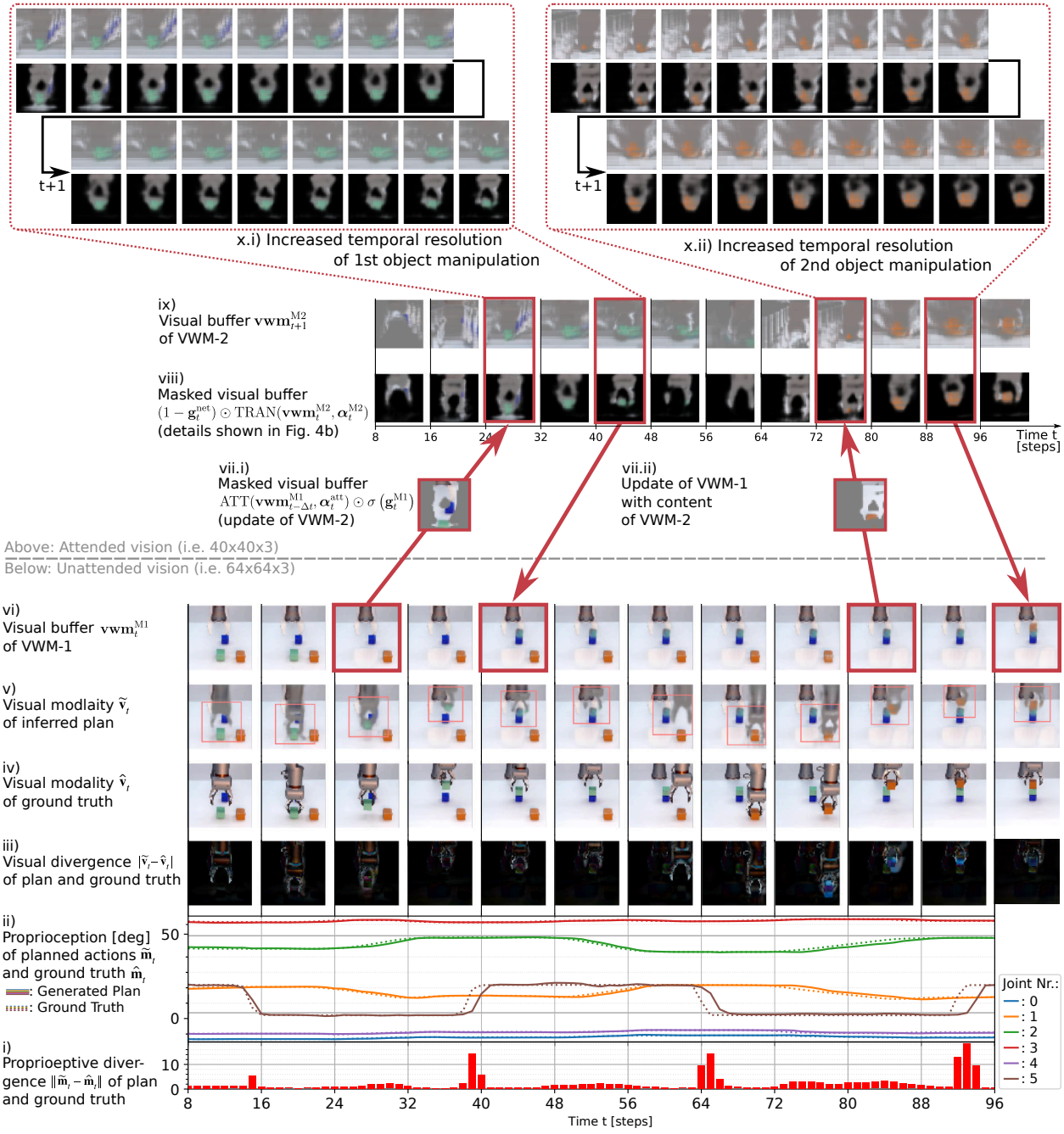


Figure 5: Results of successful planning in a case with objects having learned colors (green and blue) and a novel color (orange) with the goal of building a stack of blocks, green on blue and orange on green. Visualization of the internal states of the proposed model and the discrepancy between the inferred plan and the ground truth. An animation of the internal states is available online at https://youtu.be/pZBMEIjjh6Q.

a higher temporal resolution ($\Delta t = 1$) of the time window from time-steps 24 to 48 and from time-steps 72 to 96, respectively, in which the transformation of images stored in VWM-2 during the object manipulation can be seen.

Let us next examine in more detail what sorts of internal representations have emerged through development of internal mechanisms as a result of end-to-end learning of visuomotor samples. We can observe the same types of developments in visual attention as well as in the use of VWM-1 with those observed in the previous study (Jung et al., 2019). Parameterization of the attention transformer, generated by the RNN, results in behavior in which the focal area follows the end-effector, and thereby the manipulated object, as shown by the red annotations in Figure 5(v). Keeping the currently manipulated (i.e., moving) object in the attended region of the visual stream is beneficial to minimize the prediction error, as it cannot be represented by the static visual buffer of VWM-1. A further interpretation of the behavior of the attention transformer is that it contributes to minimization of the retinal slip, e.g., de Brouwer et al. (2001), by tracking predicted future target motion. Spatial transformer networks, such as those used for implementation of the attention transformation are difficult to train, in particular as calculations of error gradients with respect to their parameterization is based on local (i.e., bilinear) interpolation of nearest pixels. Therefore, a sufficiently high frame rate of the sequences is required to avoid sudden and noncontinuous movements of objects in visual images, which cannot be tracked by optimization of the model though back-propagation learning.

The presented results, as well as previous work by Jung et al. (2019), show that the content of VWM-1 represents static parts of the visual scene. For example, when the robot picks up the green object to manipulate its position (around time-step 16 in the ground truth shown in Figure 5(iv)), the image of this manipulated object disappears from VWM-1, seen between time-steps 16 and 40 in Figure 5(vi). However, after placement of this green object around time-step 40, the image of this placed object reappears in VWM-1 as a static image, as shown around time-step 40 in Figure 5(vi). Therefore, it is presumed that successive updates of the visual buffer of VWM-1 represent sequences of changes in the static layout of objects by capturing a meaningful structure of sub-goals in the semantics of the pick-and-place behavior.

Due to differences in basic connectivities given, and information flow between VWM-1 and VWM-2, the ways of using buffers in these two memory systems were developed very differently. At the onset of a pick-and-place action of the green block (time-step 24) toward a new position, the content of VWM-1 that represents the green block is copied into the visual buffer of VWM-2 of the network, e.g., Figure 5(vii.i). During the mental simulation of manipulating the object, VWM-2 and other connected modules retain the basic shape of the object, while they transform the visual buffer image in VWM-2 to reflect expected position and size changes in the visual appearance of the object in the mental plane.

Visualization with a higher temporal resolution from around time-steps 24 to 40, as shown in Figure 5(x.i) reveals that transformations of the content of VWM-2 represent visual imagery for manipulating the green object, in which position, as well as size changes, can be observed while mentally moving the object closer or farther away from the installed camera. Similar behavior of the network can be observed for the second block (the orange block), from around time-steps 72 to 88, as shown in Figure 5(x.ii). Although this orange block is actually a block with a novel color, information flow for manipulating this novel block is

exactly the same as with the known one, e.g., the green block shown in Figure 5(x.i).

Moreover, it can be observed, at time-step 40, that placing the green object in its final position occurs contemporaneously with a sudden update of the visual buffer of VWM-1, using the content of VWM-2, indicated by the red arrow in Figure 5(vii.ii) where we can see that the green block shown until time-step 40 in viii) is copied back to VWM-1. The same can be observed when placing the orange block at around time-step 88, as indicated by the downward arrow shown at the right side.

These observations indicate that a mental image of a continuous visual pattern for pick-and-place actions of an object on top of other objects was developed through iterative information exchanges between VWM-1 and VWM-2. In particular, it can be seen in VWM-1 that each such routine for stacking one object on another is concatenated with abstractions wherein only a static image of the 3-block layout, the result of each block manipulation, can be seen as described previously. Also, the image of the robot gripper cannot be seen in VWM-1, which is analogous to a phenomenon known in cognitive neuroscience as sensory attenuation for self-generated action (Blakemore et al., 1998, 2000). Furthermore, it can be seen that even when some objects are occluded by the robot gripper in the ground truth visual sequence, they are represented in VWM-1. This phenomenon is analogous to object permanency studied in developmental psychology (e.g., Piaget & Cook (Ed.), 1952; Baillargeon et al., 1985). Related to this, Lang et al. (2018) showed in a synthetic robotic study that sensory inputs generated by one's own movements can be diminished for the purpose of reducing possible occlusion generated by the robot's own body because sensory inputs can be mentally imaged by means of prediction, using the motor efference copy. Also, Bechtle et al. (2016) showed that object constancy can be developed by learning forward relationships between movements of robots and their sensory consequences, perceived from visual input. By looking at neural activity in VWM-2, it can be seen that detailed visual spatio-temporal patterns for each block stacking routine are generated after receiving the initial image of the object to be manipulated as copied from buffer VWM-1 with adequate attention. It should be interesting to observe that such cognitive mechanisms of chunking and abstraction can emerge through dynamic interactions among multiple sub-modules in the network, as the result of iterative end-to-end learning.

4.3.2. Comparison with Previous Models

The previous model by Jung et al. (2019) is limited to using one type of memory, which is represented by VWM-1 in the current model, by which background information, such as an image of a static object layout is preserved. A set of plan generation experiments was conducted using the current model, but excluding VMW-2, so as to demonstrate the benefits of the current model, which integrates VWM-1 and VWM-2. An analysis of the resulting visuomotor plans for new situations reveals that as expected, the model lacking VWM-2 is unable to cope with objects having novel colors in all situations. When a block with a novel color is introduced, it is treated as background and is represented correctly in VWM-1, as long as the block is not manipulated. However, as soon as this block is grasped, the RNN module, instead of VWM-1, starts to generate sequences for transforming the image of the block, wherein the color of the block gradually shifts to a known color. Generated actions are not necessarily affected by this failure in generating the correct visual appearance of the blocks; therefore, reasonable motor plans can sometimes be generated. However, in such

situations, the likelihood of confusing the order of block stacking increases. Further details on these experiments are discussed in Appendix E.

We conducted further experiments that highlight differences in representations of the task in the attended visual channel between the previous model (Jung et al., 2019) and our proposed model architecture. These results are presented in Appendix F in detail and show that unification of the dorsal and ventral visual streams and the additional VWM-2 result in a more abstract representation of the task in which unimportant details of the task are attenuated.

4.3.3. Generalization Capabilities and Content-Agnostic Information Processing

Results in Figure 5 indicate that the model is able to generate adequate plans to achieve specified goal states even when manipulating objects with unrecognized colors by achieving generalization in learning. In particular, emergence of meaningful information flow was detected where visual imagery for manipulating unseen objects is generated by network-wise visual image processing using two visual working memory modules. More specifically, when an unseen object is grasped for manipulation, its appearance is copied from VWM-1 into VWM-2. Then, an image of its being moved and placed on a specific block is generated by iterative application of an affine transformation to the copied image wherein parameters for the transformation are adequately controlled by the RNN module. In these processes, skills for transforming a given image are acquired independently of the image itself, i.e., color of objects.

What we see here is dissociation of content from information processing, since the image of the objects and its transformation is represented in different sub-modules of the network. Figure 6a illustrates the self-organized mechanism for manipulation of VWM-2 of the model. The content of VWM-2 can be updated by a memory transfer from VWM-1 (shown in red) or transformation of its own content (shown in black). Transformation of memory content is performed by an affine transformation, parameterized with low-dimensional parameters that are dynamically controlled by the RNN (dashed, black). This transformation is not performed directly on the content of VWM-2, but by temporal mapping of each pixel position in the retinotopic visual coordinate in VWM-2 in the current time-step to one in the next time-step. This means that gray scale RGB information stored at each pixel position at a current time-step is copied to a new pixel position as computed by the parameterized affine transformation for the next time-step. This might be analogous to the difference between mapping of content and mapping of its address indicated by the pointer where the content is stored. Readout and update of memory content are performed by pixel-wise masks, also generated by the RNN (dashed, red). As a result, the visual image is not predicted by the RNN directly. The RNN only determines from where and how memory content is processed. The resulting content-agnostic information processing is illustrated in Figure 6b. It shows that the focused region of the visual image includes important elements of the current sub-task (a gripper and a manipulated block) and that its pixel-wise image content is transformed over multiple time-steps in VWM-2. This is different from the case in which an RNN generates spatio-temporal patterns of visual imagery directly, as in a case without using VWM-2, since the RNN learns to generate image contents and their transformation in a mutually dependent manner, as embedded in the distributed synaptic weights. Therefore, an RNN by itself cannot transform unlearned image contents adequately.

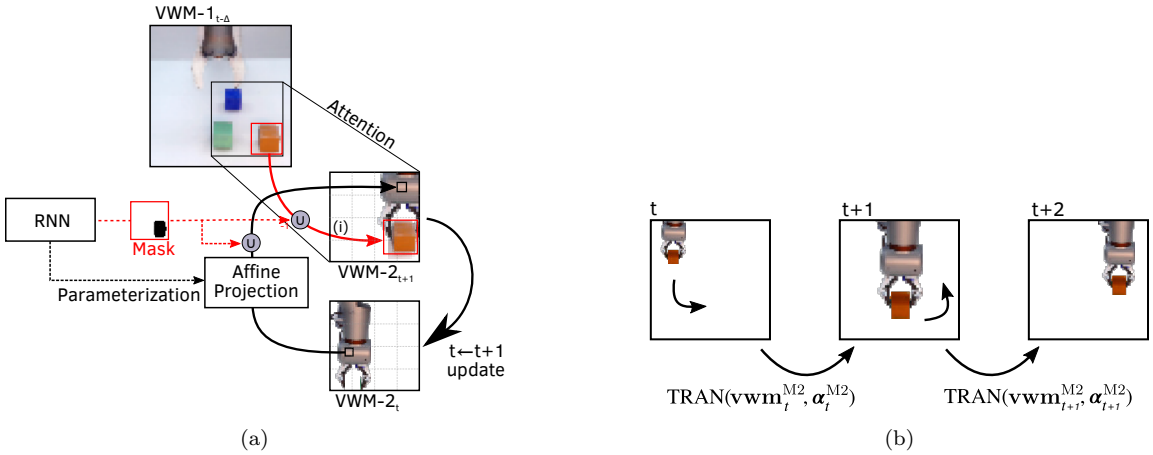


Figure 6: Illustration of observed pixel-wise image manipulation in VWM-2. The content of VWM-2 is updated by a pixel-wise copy from VWM-1 (red) or a pixel-wise transformation of its content (black) as shown in (a). Content is updated by the parameterization of an affine transformation, generated by the RNN (black; dashed line). Illustration of the sequential transformation of VWM-2 is depicted in (b).

Heuristically, the observed content-agnostic information processing, such as separation of context (i.e., motion) from content (i.e., the content of pixels), can be regarded as a generic form of factorisation of a generative model. This canonical factorisation is seen in terms of *what* and *where* pathways in the brain and may reflect the fact that knowing what something is does not tell you where it is or how it is moving. This means that one can assume conditional independence, thereby greatly increasing the efficiency of inference. This is a key aspect of variational free energy minimisation, that is defined by the factorisations (a.k.a., mean field approximations) implicit in generative models.

The current study shows that content-agnostic information processing, as described above, is crucial to achieve generalization that allows handling of previously unseen content. Next, we provide quantitative evidence to support this hypothesis.

4.4. Quantitative Evaluation

In the following section we present results of a quantitative evaluation of three different models, including those with no visual working memory, those with one visual working memory (VWM-1), and those with two visual working memory modules (VWM-1 and VWM-2) on goal-directed planning in the block stacking scenario, as described in Section 4.1. In this evaluation, errors between the ground truth trajectory and that generated by goal-directed planning are examined separately under two conditions: 1) evaluation of a test set using only 3 objects with learned colors; 2) evaluation on a test set using only objects with unlearned colors. In the first case, test trials for goal-directed planning were conducted with objects having only known colors. The test was conducted for novel initial object arrangements as well as goal arrangements using only objects of the three learned colors. In the second case, the same test trials, but using only objects with novel colors (using augmented data) were evaluated. We expect that only the model with two VWMs is capable of handling the second test case successfully, as we have explored previously in the descriptive evaluation.

The conducted evaluations confirmed our expectations, as can be seen in Figure 7. Each plot in this figure shows how the mean square error in test trials changes as the training

epoch increases for each test condition.

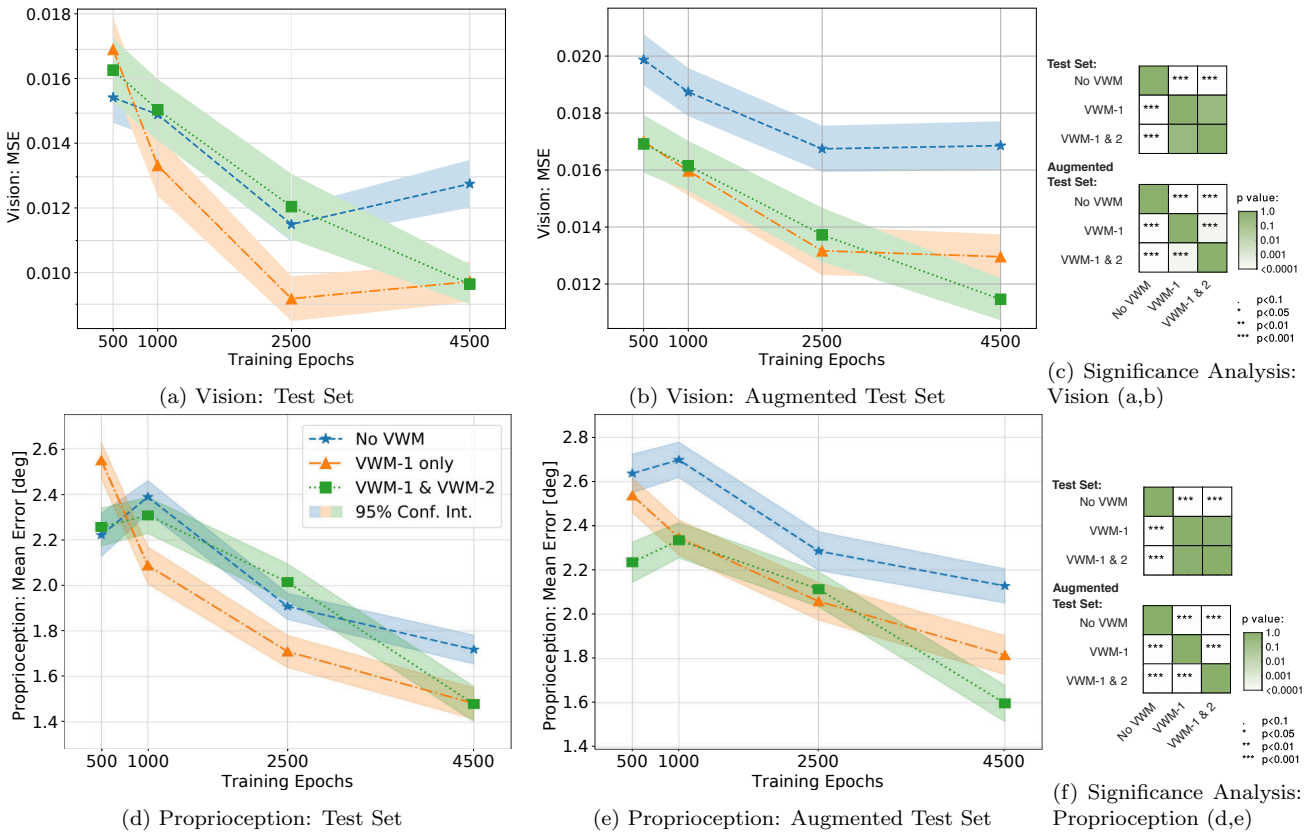
Figure 7a-b show the results in the vision channel for three models in the case of using objects with the 3 learned colors and the one using objects with novel colors, respectively. Figure 7d-e show the same for the proprioception channel for cases with 3 learned colors and with novel colors, respectively. It can be seen that the error decreased during the training epochs, except for the vision channel computed in cases using no visual working memory model and one visual working memory model in the period from epoch 2500 to epoch 4500. Comparison of the performance of the models on the test set that includes only learned colors (Figure 7a & 7d) shows that the models incorporating at least one VWM show similar performance in case a moderate generalization is required. Further, this shows that the generalization performance of the proposed extended model is not hindered by its increased complexity. Most importantly, at the end of training at epoch 4500, both errors in the vision and proprioceptive channels in the model with two visual working memory modules and in cases in which strong generalization is required (Figure 7b & 7e) are significantly smaller than in the model using one visual working memory module. Comparison of the performance in a Figure 7c and Figure 7f show a pairwise significance analysis of the reconstruction error of the visual channel and the proprioceptive channel, respectively, on the test data set. It can also be seen that test errors at the end of training for the case using no visual working memory model are significantly larger the those observed with the other two models.

These results confirm that models using visual working memory outperform the one with no visual working memory whenever strong generalization is required. A the same time, the results show that the generalization performance dos not suffer from an increased model complexity in case only moderate generalization is required. Moreover, the model with two visual working memory modules, VWM-1 and VWM-2, outperforms the one using only VWM-1 in the goal-directed planning task, which requires generalization for novel situations dealing with objects with novel colors.

5. Discussion

This study investigated a certain class of generalization problems involving context-agnostic information processing, which both humans and artificial agents encounter in routine action generation, by conducting synthetic neurorobotic experiments in simulation. More specifically, we examined how robots are able to generate goal-directed action plans in object manipulation by learning even with unfamiliar objects having novel features such as color, by adequately generating sequential mental images for manipulating them.

For this purpose, we revisited our previous study (Jung et al., 2019) and conducted extended simulation experiments. The current study used a complex network with synergy between a set of sub-module neural networks, including multiple visual working memory modules (VWMs), a visual attention module, an executive network for prediction of motor and visual images, and controls for visual attention and masking of the visual images in the VWMs. One essential update from the previous model (Jung et al., 2019) is that the current model employs an additional VWM and considers further connectivity between this module and others. This is to evaluate our main hypothesis that generalization required for content-agnostic information processing can be achieved if the whole network can adequately incorporate this additional VWM via interactions through learning from past experience.



(a) Vision: Test Set

(b) Vision: Augmented Test Set

(c) Significance Analysis: Vision (a,b)

(d) Proprioception: Test Set

(e) Proprioception: Augmented Test Set

(f) Significance Analysis: Proprioception (d,e)

Figure 7: Discrepancy error between the inferred plan and the ground truth in two cases, namely dealing with objects with learned colors and one with novel colors. Authors: Figure description is corrected.The visual channel (a-c) and proprioceptive channel (d-f) are evaluated separately. Evaluation in the test with learned objects and novel objects is shown in addition to the pairwise evaluation of the significance at the end of training (4500 epochs). Models without visual working memory, with one visual working memory module (VWM-1), and with two visual working memory modules (VWM-1 & VWM-2) memory are compared.

Learning of the whole network is accomplished by means of free energy minimization (Friston, 2005) by following the predictive coding formula (Mesulam, 1998; Rao & Ballard, 1999) in end-to-end learning of sampled visuo-proprioceptive trajectories.

After learning, we evaluated the performance of the model network in generating goal-directed action plans using active inference (Friston et al., 2006), in cases that involved manipulating blocks with novel colors. The results showed a significant improvement in performance when using an additional VWM, compared to a case using only a single VWM. A detailed analysis of whole network activity first revealed that when the robot grasps a block to move it, visual attention follows the block autonomously while static blocks behind a manipulated block are retained in the first VWM. This is the same as observations in Jung et al. (2019). More interestingly, the attended visual image of the grasped object was copied into the newly introduced VWM, and it was spatially transformed to generate image sequences of stacking objects by following the generated visual plan. This phenomenon was observed during manipulation not only for blocks with learned colors, but also for blocks with previously unseen colors.

The aforementioned analytical result reveals how content-agnostic information processing was developed in the course of learning while dealing with generalization for mental simulation of objects with novel colors. The essential aspect of the mechanism acquired through learning is dissociation of visual image contents from the mechanism for their manipulation. In the model after learning, an image of the object to be manipulated is saved once in VWM-2. Then, to achieve a specified goal, the image is transformed by means of temporal mapping of each pixel position in the visual coordinate of VWM-2 as controlled by RNNs in the executive network. By this means, an image once saved in VWM-2 can be transformed regardless of its content, because transformation is conducted with the position of each pixel, independent of the content (RGB information) saved at each pixel. However, without VWM-2, this sort of dissociation cannot be achieved and generalization for unlearned object images cannot be expected, since RNNs alone cannot generate transformation of novel object images in a content-agnostic way.

The experimental design as presented in Section 4 aims at specific cases of the generalization problem that involve noisy real world data, manipulating objects with unlearned locational distributions, compositionality through planning for unlearned tower configurations and representation of objects with unlearned colors.

These experiments show that strong generalization can be achieved and that content-agnostic information processing of color information is developed in the model during training. To further support our hypothesis of content-agnostic information processing through utilization of an additional VWM (VWM-2), as illustrated in Figure 6, we conducted additional experiments involving more complex visual representations. In Appendix G, results of additional experiments that require content-agnostic processing of objects with unlearned textures are shown. For these experiments we rearranged the task scenario as described in Section 4 and restricted task complexity to maximize the size of objects in relation to image resolution. Further details of experimental conditions are listed in Appendix Appendix G. A modification of the task scenario was necessary to reduce computational resource requirements by limiting the number of training samples and reducing the sequence lengths.

The quantitative and qualitative analysis of the additional experimental results support our previously stated hypothesis of content-agnostic visual processing and show that our

newly proposed model with two VWMs achieves superior generalization in generating goal-directed visuo-motor plans dealing with objects with previously unseen textures. As shown in the example visualizing internal neural activities of the model (Figure G.14a), it turned out that the ways of using two VWMs are compatible to those shown in the main experiment described in section Section 4.

A future study should examine how the current scheme of content-agnostic information processing using an additional VWM (VWM-2) can be applied to generalization in learning with more complex situations, such as manipulating objects with novel shapes and sizes. Generalization with different shapes and sizes of the objects is expected to be more challenging, because such situations obviously will involve adaptation of motor controls if those variations affect the means of manipulating those objects. This is not only a problem of context-agnostic information processing in the visual pathway, but should involve generalization problems in the motor pathway. This issue will be examined with deliberative experiments in the future.

Problems involving moving objects and dynamically changing environments are not addressed in the current study, which assumed that goal-directed plan generation is performed only in a static environment. If distractor objects move, such situations can be resolved if the model network can learn to ignore those objects. This could be achieved even with the current model if the visual attention module functions adequately. If objects to be manipulated move, this situation can also be resolved if the model network can learn to predict how the objects move. These examinations are left for future studies.

Although the current study showed that the proposed neural network architecture using two VWMs exhibited competitive performance in terms of the discrepancy between inferred plan trajectories and ground truth trajectories, our preliminary study showed that the success rate of each task by executing the inferred motor plans with a real robot could not exceed 50%. The resultant low performance is due to the fact that relatively low pixel resolution (64x64) in the video image was used in the current experiment, because of the excessive computational cost for inference through video frames. A one-pixel prediction error in the size of an object image could result in up to a 5-cm position error of the robot gripper when grasping objects, a huge error considering the 5-cm block size. Future studies introducing depth information, as well as larger image resolution, such as 256x256 in the video, along with development of an effective parallelization scheme in inference of plan generation through the video frames, should improve the success rate greatly.

Furthermore, extended studies should investigate how the model could deal with the problem of online planning in a physical setting. This should require the model to adapt to dynamically changing environments in real time. For this purpose, the model should be extended such that it can cope with the following three issues. First, the robot should be able to recognize the current situation by maximizing the evidence lower bound. Second, it should be able to update current goal-directed plans based on its currently recognized situation by maximizing the estimated lower bound. Third, it should be able to act on the environment by executing a plan to be carried out in real time. The retrospective and prospective inference scheme (REPRISE) proposed and investigated by Butz et al. (2019) represents a good starting point to consider this problem.

6. Acknowledgements

This work was supported by JSPS KAKENHI Grant Number JP20K19901. We are grateful for the help and support provided by the Scientific Computing and Data Analysis section of the Research Support Division at OIST.

Appendix A. Implementation Details of Attention Transformer

Implementation of the attention transformer ATT is based on a spatial transformer network (STN; Jaderberg et al., 2015). The STN performs a pixel-wise spatial transformation of an input image with the same transformation performed for each color-channel. The transformation is defined by a grid generator $\mathcal{T}_{\boldsymbol{\alpha}_t^{\text{att}}}(G_i) = \begin{bmatrix} f(x_i,[\boldsymbol{\alpha}_{t,3}^{\text{att}},\boldsymbol{\alpha}_{t,2}^{\text{att}},\boldsymbol{\alpha}_{t,1}^{\text{att}}]) \\ f(y_i,[\boldsymbol{\alpha}_{t,4}^{\text{att}},\boldsymbol{\alpha}_{t,2}^{\text{att}},\boldsymbol{\alpha}_{t,1}^{\text{att}}])) \end{bmatrix}$, defining a vector field that maps transformed image coordinates $G_i = (x_i, y_i)$ of a regular grid $G = \{G_i\}$ to the input space. A regular grid of 2D pixel positions in the output image is mapped by application of $\mathcal{T}_{\boldsymbol{\alpha}_t^{\text{att}}}$ to its corresponding input positions before a bilinear differentiable sampling kernel, as in Jaderberg et al. (2015), is applied. The inverse transformation ATT^{-1} is performed by application of $\mathcal{T}^{-1}_{\boldsymbol{\alpha}_t^{\text{att}}}$ and simultaneous rescaling is achieved by adapting the dimensionality of the regular grid that is used for the sampling process accordingly. Mapping of pixel coordinates is performed in such a way that a combined representation of dorsal and ventral image information is possible, as illustrated in Figure A.8. Independent projections of pixel coordinates for each image dimension are based on a mixture of linear transformations l_1 to l_3, as described by:

$$f(x, \boldsymbol{\theta}) = \sigma_A(x, \boldsymbol{\theta})l_1 + \sigma_B(x, \boldsymbol{\theta})l_2 + (1 - \sigma_A(x, \boldsymbol{\theta}) - \sigma_B(x, \boldsymbol{\theta}))l_3 \tag{A.1}$$

The interpolation by σ_1 and σ_2 between the focal and peripheral magnification levels is defined as

$$\sigma_A(x, \boldsymbol{\theta}) = \sigma((1 - \theta_2)\theta_1 - x) \quad \text{and} \tag{A.2}$$
$$\sigma_B(x, \boldsymbol{\theta}) = \sigma(x - (\theta_1 + \theta_2 - \theta_1\theta_2)), \tag{A.3}$$

with $\boldsymbol{\theta} \in \mathcal{R}^3$ defining the transformation along each image dimension independently: θ_1 defines the center of the focal area, θ_2 defines the relative size of the focal area, and $1/\theta_3$ determines the zoom factor inside the focal area. The interpolation can be based on a sigmoidal or a Heaviside step function. The linear transformations l_1 to l_3 are defined as:

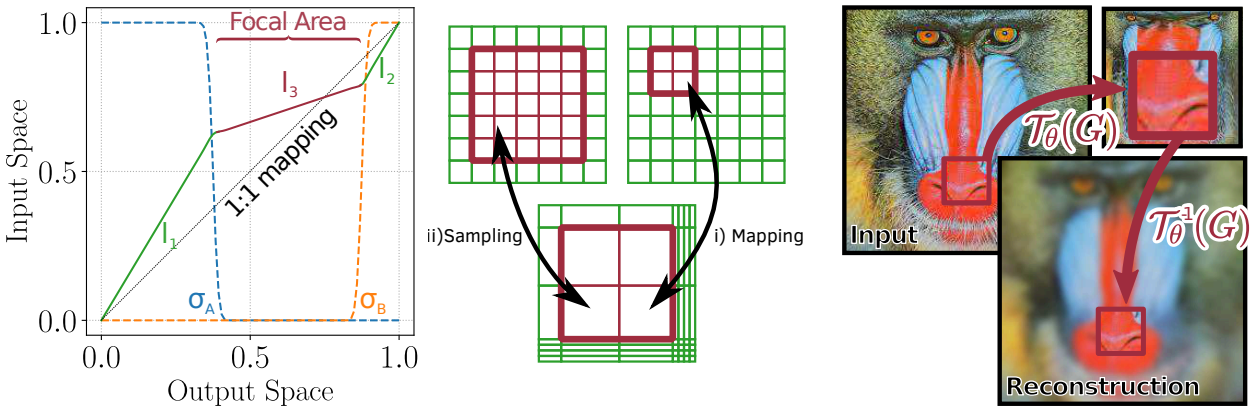
$$l_1 = \frac{x(\theta_1 - (\theta_1\theta_2\theta_3))}{\theta_1 - (\theta_1\theta_2)}, \tag{A.4}$$

$$l_2 = (1 - \theta_1 - (1 - \theta_1)\theta_2\theta_3)\frac{x - \theta_1 - (1 - \theta_1)\theta_2}{1 - \theta_1 - (1 - \theta_1)\theta_2} \tag{A.5}$$

$$+ \theta_1 + (1 - \theta_1)\theta_2\theta_3 \quad \text{and} \tag{A.6}$$

$$l_3 = (x - \theta_1)\theta_3 + \theta_1. \tag{A.7}$$

In applying the Heaviside step function, the back transformation $\mathcal{T}^{-1}_{\boldsymbol{\alpha}_t^{\text{att}}}$ can be estimated trivially by the partial inverse of the linear functions. In applying a sigmoidal transition



(a) (b) (c)

Figure A.8: Visualization of the spatial transformer for *fovea-like* visual attention. The grid transformer $\mathcal{T}_{\boldsymbol{\alpha}_t^{\text{att}}}(G)$ and its inverse are based on (a) the transformation $f(\boldsymbol{\theta})$. (b) depicts an illustration of the pixel wise mapping (i) between output and input image panels and the successive sampling (ii) of pixel data. An example result of the described image transformation is shown in (c).

between the linear functions, we refer to an approximate inverse function estimation, if an analytical solution cannot be found. In this case, the inverse transformation is described by the following equation:

$$f^{-1}(\boldsymbol{\theta}) = \sigma_A^{-1}(\boldsymbol{\theta})l_1^{-1} + \sigma_B^{-1}(\boldsymbol{\theta})l_2^{-1} + (1 - \sigma_A^{-1}(\boldsymbol{\theta}) - \sigma_B^{-1}(\boldsymbol{\theta}))l_3^{-1}, \tag{A.8}$$

with

$$\sigma_A^{-1}(\boldsymbol{\theta}) = \sigma((1 - \theta_2\theta_3)\theta_1 - x) \quad \text{and} \tag{A.9}$$
$$\sigma_B^{-1}(\boldsymbol{\theta}) = \sigma(x - (\theta_1 + \theta_2\theta_3 - \theta_1\theta_2\theta_3))). \tag{A.10}$$

Appendix B. Implementation Details of Visual Loss Function

Our previous studies showed that back-propagated visual error signals suffer from over-representation of the focal area caused by the variable pixel densities of the attention transformation. To balance the contribution of the visual errors that originate in the focused and unfocused regions of the generated visual output of the model, $\mathbf{s}_t^{\text{att}}$ was introduced to perform a scaling of the error signal with respect to the distance to the center of the focal area. The estimation of $\mathbf{s}_t^{\text{att}}$ is performed by the following calculations:

$$\mathbf{s}_{t,ij}^{\text{att}} = \alpha_{\text{s}} \left(\begin{bmatrix} \alpha_{t,3}^{\text{att}} \\ \alpha_{t,4}^{\text{att}} \end{bmatrix} - \begin{bmatrix} i/\mathrm{N_{in}} \\ j/\mathrm{N_{in}} \end{bmatrix} \right)^2, \tag{B.1}$$

for image resolution $(\mathrm{N_{in}}, \mathrm{N_{in}})$ in order to scale the visual error signal in relation to the distance to the center of the focal region.

Appendix D. Example Sequences of Robotic Data Sets

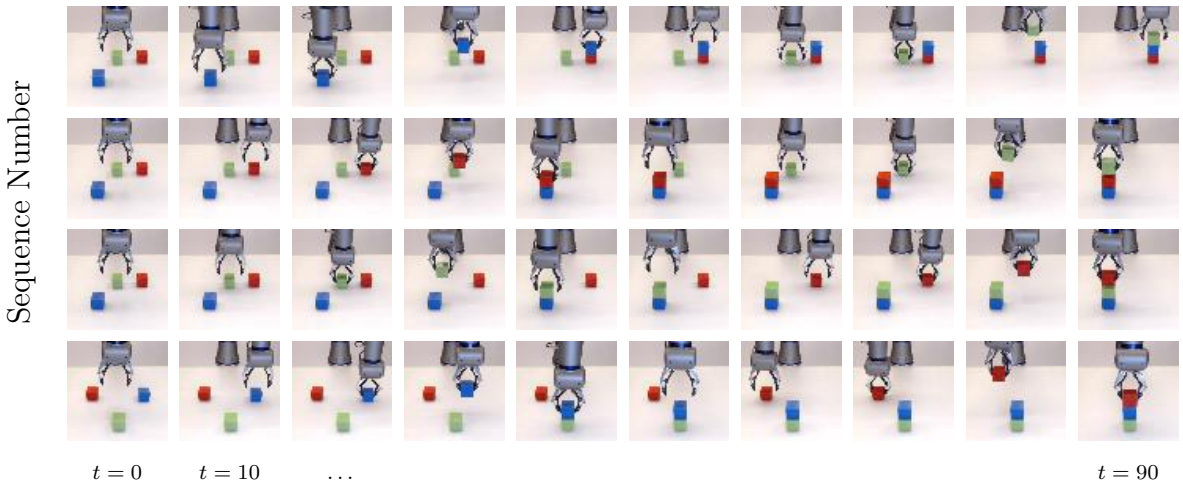


Figure D.9: Example sequences of the robotic dataset. Three colored cubes (red, green and blue) are in the recorded dataset. Positioning of the blocks in the workspace is based on a 10x10 grid for training and an 8x8 grid for testing. Permutation of the stacking order results in six final tower configurations (RGB, RBG, GBR, GRB, BGR and BRG), whereas the last two configurations (BGR and BRG) are excluded from the training set.

Appendix C. Training & Planning Procedure

Training and planning are performed according to the pseudocode in Algorithm 1 and Algorithm 2, respectively.

Algorithm 1: Training procedure

initialization (prior and posterior):
$(\boldsymbol{\mu}_0, \boldsymbol{\sigma}_0) = \boldsymbol{\theta} \leftarrow (0, 1)$;
$(\boldsymbol{\mu}_0^g, \boldsymbol{\sigma}_0^g) = \boldsymbol{\phi}^g \leftarrow \boldsymbol{\theta} \quad \forall g \in \mathcal{D}_{\text{train}}$;
for $e \leftarrow 1$ **to** N_{epochs} **do**
  for $g \leftarrow 1$ **to** $N_{samples}$ **do**
    sampling:
    $\mathbf{s}_0^g \leftarrow \boldsymbol{\mu}_0^g + \boldsymbol{\epsilon} \odot \boldsymbol{\sigma}_0^g$;
    generation:
    $(\widetilde{\mathbf{v}}^g, \widetilde{\mathbf{m}}^g) \leftarrow FWD((\mathbf{v_0}^g, \mathbf{m_0}^g), \mathbf{s}_0^g, \mathbf{w}_{\text{net}})$;
    loss calculation:
    $l_g \leftarrow L^g(\widetilde{\mathbf{v}}^g, \widetilde{\mathbf{m}}^g, \boldsymbol{\phi}^g, \boldsymbol{\theta})$;
    gradient descent:
    $(\mathbf{w}_{\text{net}}, \boldsymbol{\theta}, \boldsymbol{\phi}^g) \leftarrow Adam(\partial_{\mathbf{w}_{\text{net}}} l_g, \partial_{\boldsymbol{\theta}} l_g, \partial_{\boldsymbol{\phi}^g} l_g)$;
  end
end

Algorithm 2: Planning procedure

initialization (posterior):
$(\boldsymbol{\mu}_0^g, \boldsymbol{\sigma}_0^g) = \boldsymbol{\phi}^g \leftarrow \boldsymbol{\theta} \quad \forall g \in \mathcal{D}_{\text{test}}$;
for $g \leftarrow 1$ **to** $N_{samples}$ **do**
  for $e \leftarrow 1$ **to** N_{epochs} **do**
    for $r \leftarrow 1$ **to** $N_{runs} = 16$ **do**
      sampling:
      $\mathbf{s}_0^g \leftarrow \boldsymbol{\mu}_0^g + \boldsymbol{\epsilon} \odot \boldsymbol{\sigma}_0^g$;
      generation:
      $(\widetilde{\mathbf{v}}^g, \widetilde{\mathbf{m}}^g) \leftarrow FWD((\mathbf{v}_0^g, \mathbf{m}_0^g), \mathbf{s}_0^g, \mathbf{w}_{\text{net}})$;
      loss calculation:
      $l_r^g \leftarrow L_p^g(\widetilde{\mathbf{v}}^g, \widetilde{\mathbf{m}}^g, \boldsymbol{\phi}^g, \boldsymbol{\theta})$;
    end
    gradient descent (best run):
    $r_{\text{best}} \leftarrow \arg\min_r l_r^g$;
    $\boldsymbol{\phi}^g \leftarrow Adam(\partial_{\boldsymbol{\phi}^g} l_{r_{\text{best}}}^g)$;
  end
end

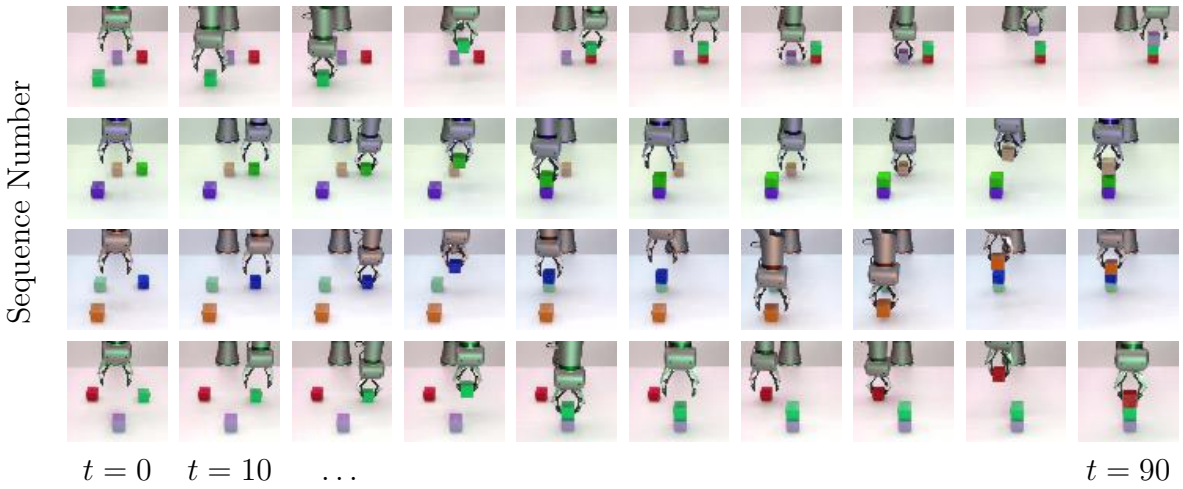


Figure D.10: Example sequences of the augmented robotic dataset. Random permutations in the color planes of the original dataset (Figure D.9) result in block colors not present in the training set. The dataset used to evaluate model performance in cases of strong generalization is required, since typically, representation of previously unseen objects is challenging for RNNs.

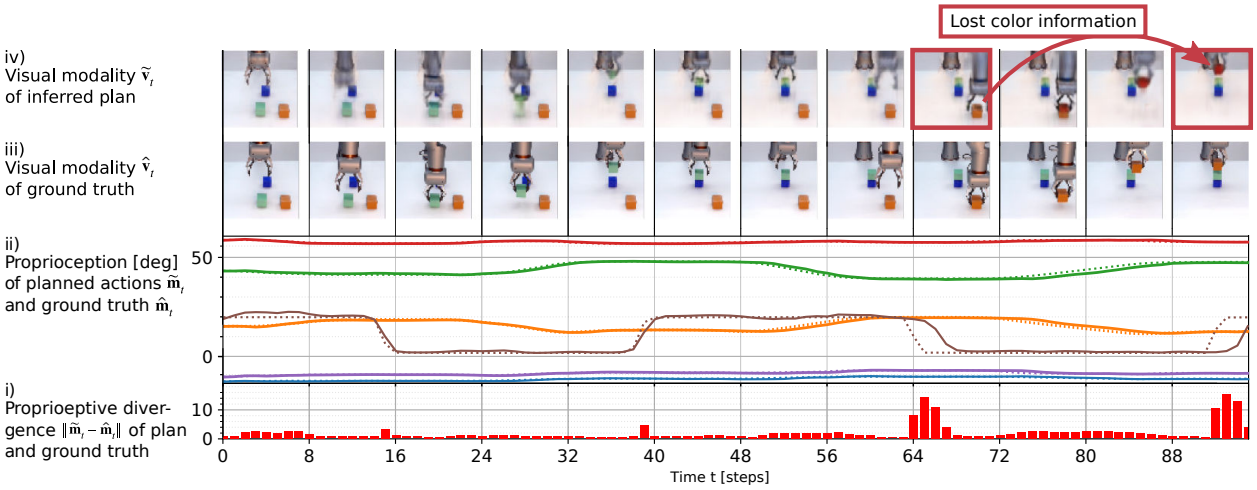
Appendix E. Comparison with Previous Models

Figure E.11 shows two examples in which the current model without VWM-2 fails to generate correct visuomotor plans as the color information was lost. In the first example, Figure E.11a, the color information of the blocks is lost and block colors converge, i.e., the orange color of the manipulated object is replaced by red. Nevertheless, the generated trajectory still achieves reasonable accuracy as block positions are approached in the correct order.

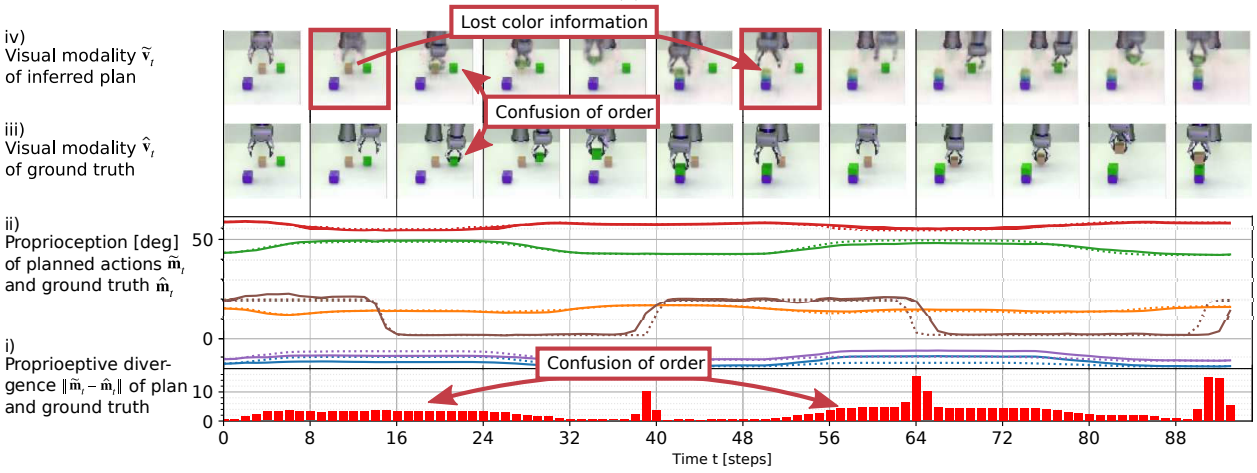
The second example shows an inferred plan that loses the correct color information, i.e., a peach-colored manipulated block becomes green, as shown in time-steps 8 to 48 of Figure E.11b(iv). The lost color information results in confusion of the order of the stacking operations, and results in a large discrepancy in vision and proprioception channels between the inferred plan and the ground truth. As shown in Figure E.11b(i), the discrepancy in the proprioceptive channel is high when an object is picked from the table (around time-steps 16 and 64) and is low (neglecting variability in the timing of actuation of the gripper) while placing the objects in a tower configuration (around time-steps 40 and 88). The above-mentioned fluctuation of the discrepancy between the inferred plan and the ground truth is caused by a plan that still succeeds in building a tower configuration at the position specified by the current goal, but confuses the color information and the order of two consecutively manipulated blocks.

Appendix F. Explicitly Decomposed Representation

This section presents an evaluation of our current approach versus previous models that utilize dorsal and ventral visual streams (Jung et al., 2019). Figure F.12 depicts a comparison of the internal visual representations in the convolutional RNN blocks and the final visual prediction for one successful goal-directed action plan. Three model architectures are compared: the previous model with two visual streams (Figure F.12a), our newly proposed model



(a)



(b)

Figure E.11: Visualization of failed planning attempts, as observed in a model with only one visual working memory VWM-1. (a) and (b) show such two examples. The RNN is not able to generalize to new colors and loses color information during manipulation of the blocks. The loss in color information results in confusion of the stacking order, as evidenced by a high prediction error of the proprioceptive modality.

that utilizes only one visual stream, and one VWM (Figure F.12b), and our proposed model including two VWMs (Figure F.12c). When two visual streams are utilized, as shown in Figure F.12a, the color information of all blocks is represented simultaneously in the generated output of the convolutional RNN blocks of the peripheral and ventral pathways of the model. This means that not only the currently modified object can be identified, but also the other objects that are not relevant to the currently ongoing sub-task. This indicates that RNN blocks represent the complete task configuration throughout all layers. In comparison, the organization of the model architecture into one unified visual stream, shown in Figure F.12b, results in a more explicitly decomposed representation. In this case, the background and the unmodified objects are represented in VWM-1 and the higher-level vision network represents only images related to the currently ongoing sub-task. Figure F.12b(ii.i) depicts the visual representation of the scene after placing the blue block on the green block. In this case, predictions in the attended visual image are solely based on generated images of the RNN, as VWM-2 is not available in this experiment. Only color information of the currently manipulated blue block is represented in the visual predictions of the convolutional RNN block. Moreover, the model retains an abstracted representation of the remaining blocks in the form of an uncolored image. The results indicate that the current model is able to learn a more abstracted and compositional representation of the training data by restriction of representations to sub-task-specific information. In this case, the model represents the action of stacking a specific block (color information available) on top of an arbitrary block (no color information available).

The whole image is decomposed into the representation of the currently manipulated and other objects as background by using only the architecture of the unified visual stream shown in Figure F.12b. This is more successfully performed by adding VWM-2 into the unified visual stream architecture, shown in Figure F.12c, in which color information of the currently manipulated object is more explicitly extracted by the RNN as compared to the experiment shown in Figure F.12b.

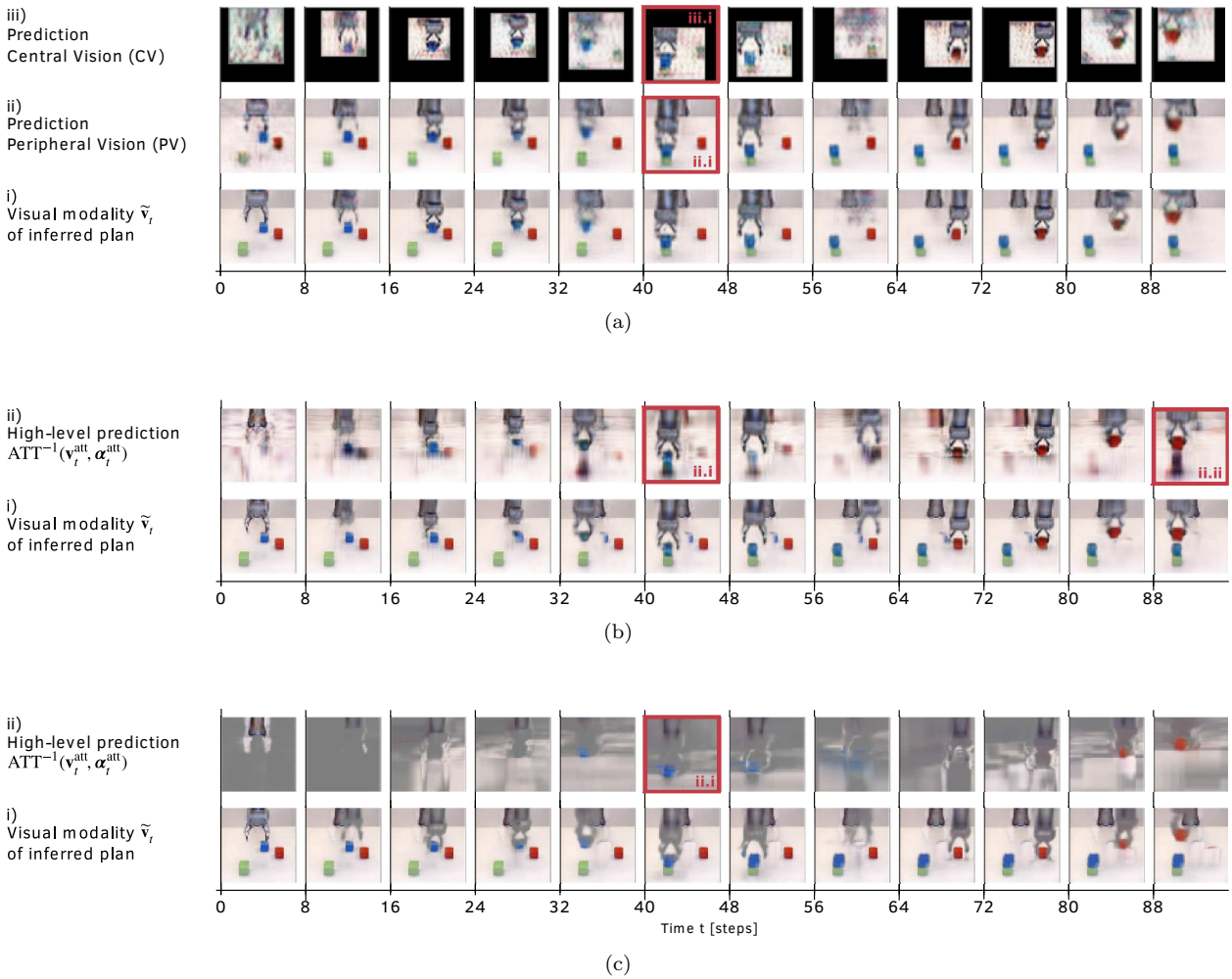


Figure F.12: Comparison of visual representations in RNN blocks for a successfully inferred plan in three different models. (a) shows predictions of the previous model (Jung et al. (2019)) that are limited to use of one type of memory and proposes two visual channels, peripheral (a.ii) and central (a.iii) vision; (b) shows predictions of the current model without VWM-2; and (c) shows predictions of the current model including both VWM-1 and VWM-2. Unification of the visual stream leads to an explicit representation of the manipulated object by the recurrent neural network, as other objects in the scene are not represented by color.

Appendix G. Experiment: Content-Agnostic Information Processing for Generalization for Unlearned Textures

The purpose of an additional experiment shown in this section is to examine whether content-agnostic information processing developed in the model could account for generalization with a different novel situation, objects with unlearned texture patterns.

For this experiment, a modified task scenario has been designed. In comparison with experiments presented in Section 4, the scenario has been redesigned such that the camera is positioned closer to the workspace of the robot in order to maximize the numbers of pixels of the visual sensation that are occupied by objects in the scene. The enlarged appearance of the objects in the visual image (approx. 12x12 pixels) allows the representation of complex patterns (textures) that are mapped on the objects in the lower-dimensional image computed through the attention transformation, i.e. the visual feature space of VWM-2 and convLSTM. In the following, the experiment design and the experimental result are described.

Appendix G.1. Task Design and Generation of Training Data

These experiments have been performed in simulation of the Torobo robot. As in the previous experiments, sequences of object manipulations are recorded. Given an initial posture of the robot, the robot is commanded to generate the following sequence of movements: 1) it reaches a specific target object; 2) grasps the object; 3) moves the object to a specific target location; 4) releases the object; 5) moves back to the final posture. The design of the task is depicted in Figure G.13a. Recording of the sequences has been automated and trajectory generation ensures that the final length of the sequence is 33 time-steps in every case. In total, 250 training sequences and 80 sequences for testing have been recorded. Four example sequences used for training are visualized in Figure G.13f. As for our previously discussed experiment, initial object locations are sampled from two different distributions, one for training and one for planning. The desired placement position of the target object, as indicated by a black square, was moved along its y coordinate on the table and kept inside the workspace of the robot. In addition to the colors red (r), green (g) and blue (b), two texture patterns have been mapped onto the objects in the scene during generation of the training dataset. The possible appearances of the objects in the training set are listed in Figure G.13b. For generation of the test data set, the colors cyan (c), yellow (y) and pink (p), as well as textures showing cross and triangle patterns have been mapped onto the objects. The unlearned colors and textures in the test dataset are depicted in Figure G.13c. In both cases, up to two randomly selected distractor objects have been placed at randomly selected positions, such that they do not interfere with movements of the robot. In addition to cubes, we introduced two additional object shapes (a cylinder and a triangular bar) as distractor objects, as shown in Figure G.13d and Figure G.13e.

Appendix G.2. Experiment

Training is conducted analogously to the previous experiment described in Section 4. After minimizing the loss function L^g (Section 3.5) for all training sequences for 4000 epochs, generalization in action plan generation to achieve novel goal states specified by the visual images is evaluated. Action plan generation to achieve novel goals dealing with variable

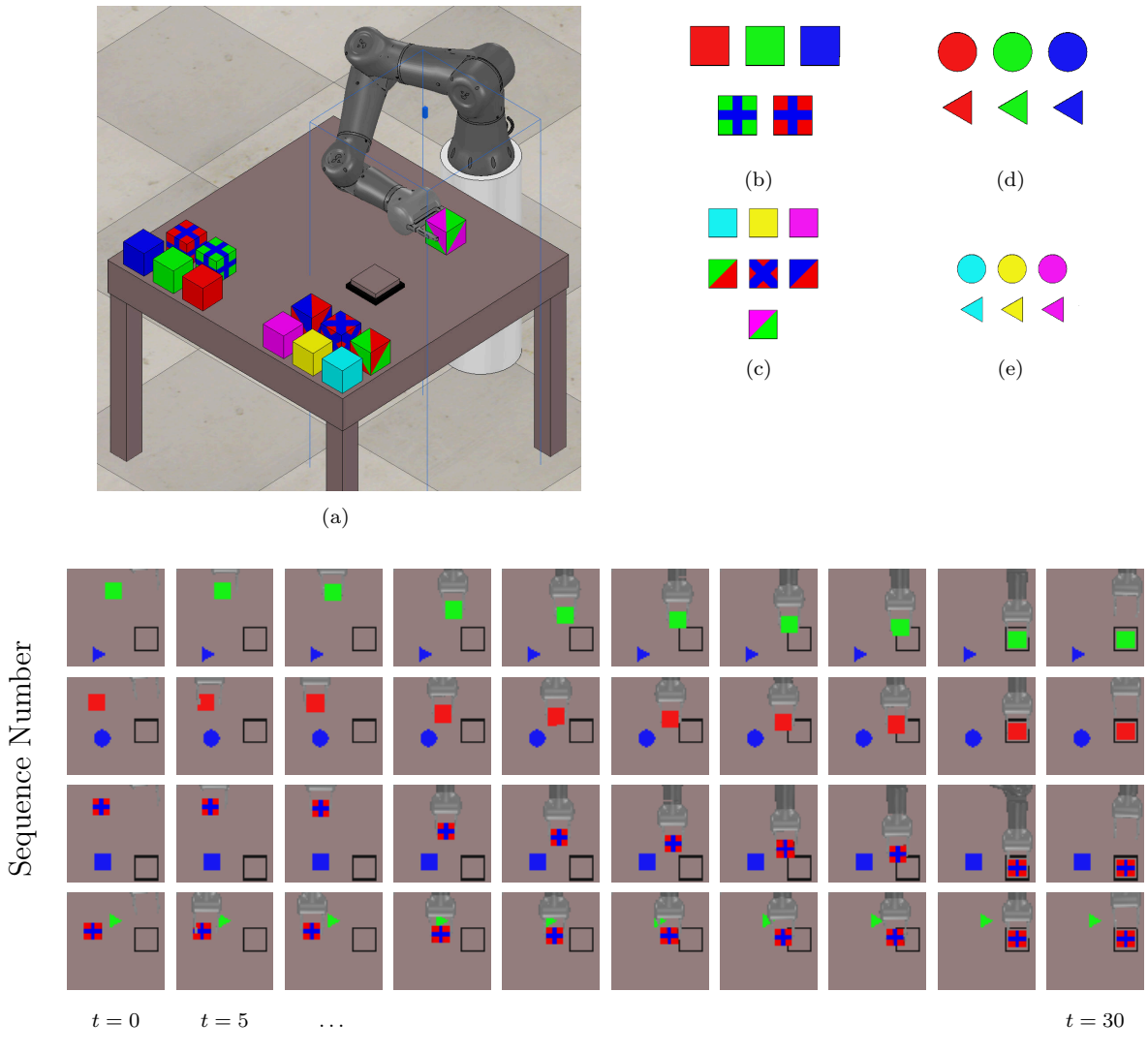


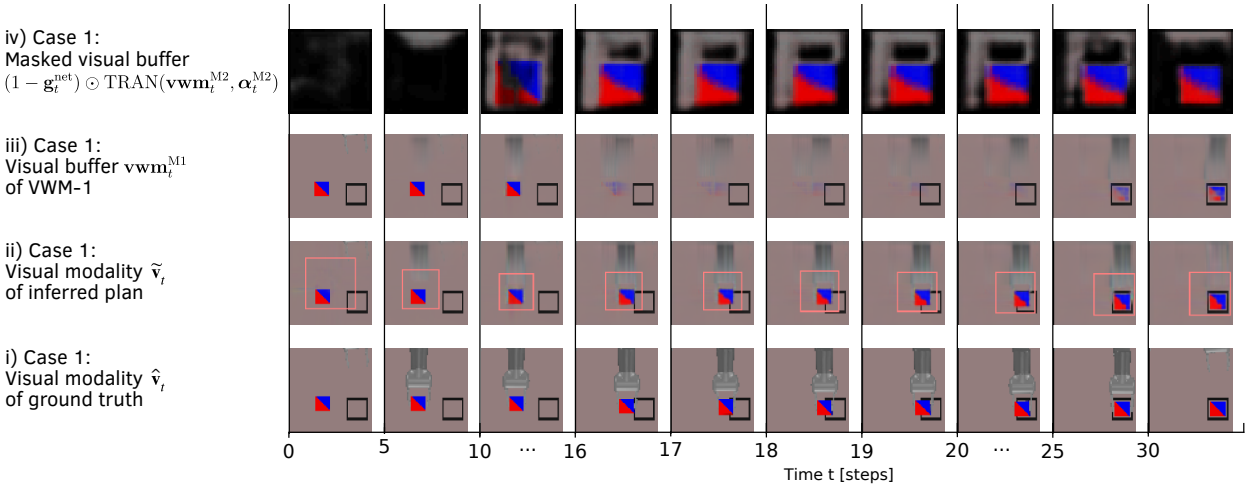
Figure G.13: Task design (a), the robot moves a block (with previously unseen color, texture & position) onto a desired target plane (indicated by black square). Visualization of the variability of the appearance of the objects is shown in (b-e): training set (b); new appearances during planning (c); disturbance objects for training (d) and planning (e) include further object shapes for assessment of generalization capabilities for previously unseen backgrounds. Example visual sequences of the training set are shown in (c).

positions both for the initial and the final object positions is performed using the same planning scheme described in Section 3.5.1. The inference for the goal-directed planning is iterated for 100 epochs.

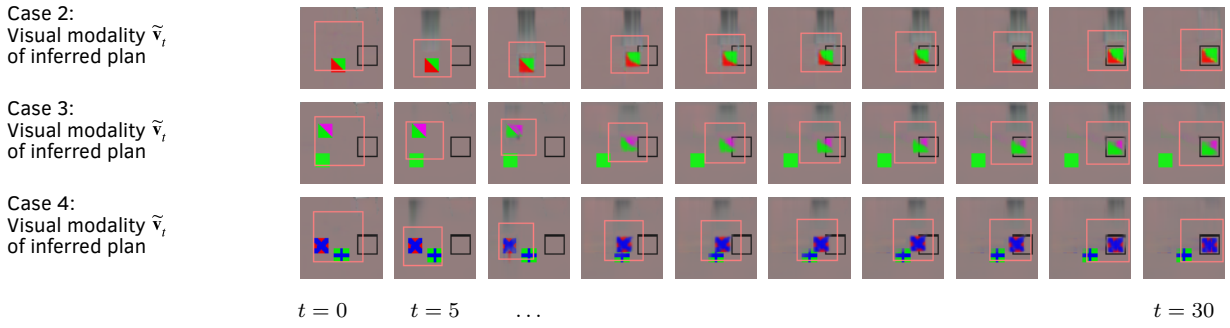
From repeated simulation results, we found that performance of goal-directed plan generation using novel texture objects is significantly better when using two VWMS compared to the case with one VWM. The model with two VWMs results in significantly lower planning loss L_p^g over all test sequences g, Its mean is 0.0174 ± 0.0010, in comparison to 0.0244 ± 0.0018 for the model with only one VWM (p>0.95). As expected, the difference in the mean squared error between both models is significant (p>0.95) as well, with error 0.0040±0.0002 for the model with two VWMs and 0.0048±0.0003 when only one VWM is utilized. The mean squared error in proprioception is slightly lower (not statistically significant) in the model with two VWMs, 0.84±0.07 [deg^2], in comparison with the model with only one VWM, 0.95±0.10 [deg^2].

Figure G.14 shows some example results of goal-directed planning using objects with novel texture patterns in cases involving the model with two VWMs and that with one VWM. Figure G.14a shows an example of successful goal-directed plan generation using the model with two VWMs wherein the contents of masked VWM-2, VWM-1, the inferred visual plan, and its grand truth are shown. We can see content-agnostic (pixel-wise) transfer of the visual appearance of objects which is analogous to that described in Section 4.3.3. In this figure, an empty red square denotes the focus area of the attention transformer and an empty black square denotes the goal position to which the object should be moved. Figure G.14b shows other three representative examples of inferred visual plans generated by the current model using two VWMs. It shows that the model can deal with different types of novel texture objects. Figure G.14c shows two representative examples of inferred plans generated by the model using only one VWM. Objects with novel textures cannot be manipulated well in the generated visual pan image. Both cases show that texture patterns of the objects are changed incorrectly during manipulation of those objects in the inferred plans.

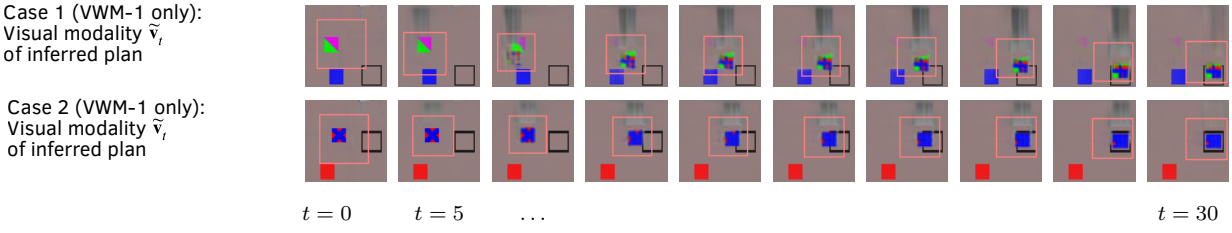
In summary, only the model with two VWMs is capable of representing manipulation of objects with previously unseen textures by copying their visual images into VWM-2 and by pixel-wise transformations of this memory. These results support our claim that content-agnostic information processing developed in the model can enhance generalization in dealing with novel situations, including cases of manipulating objects with novel textures as well as with novel colors.



(a)



(b)



(c)

Figure G.14: Planning results for previously unseen object positions, colors, and textures. Visualization of the internal states of the newly proposed model (two VWMs) is shown in (a), further exemplary visual predictions are depicted in (b). Failed planning attempts of the model limited to only one VWM are shown in (c). An empty red square depicts the focus area of the attention transformer, as predicted by the model and an empty black square depicts the goal position where the object should be moved. An animation of the internal states is available online at https://youtu.be/frp1Utxx-XA

References

Arbib, M. A. (1981). *Perceptual structures and distributed motor control*, chapter Handbook of Physiology: The Nervous System, Motor Control, (pp. 1449–1480). American Cancer Society.

Arie, H., Endo, T., Arakaki, T., Sugano, S., & Tani, J. (2009). Creating Novel Goal-Directed Actions At Criticality: A Neuro-Robotic Experiment. *New Mathematics and Natural Computation*, 05(01), 307–334.

Ba, L. J., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. *Computing Research Repository*, abs/1607.06450.

Baillargeon, R., Spelke, E. S., & Wasserman, S. (1985). Object permanence in five-month-old infants. *Cognition*, 20(3), 191–208.

Baltieri, M. & Buckley, C. L. (2017). An active inference implementation of phototaxis. *Artificial Life Conference Proceedings*, 14(29), 36–43.

Baltieri, M. & Buckley, C. L. (2019). Pid control as a process of active inference with linear generative models. *Entropy*, 21(3).

Bechtle, S., Schillaci, G., & Hafner, V. V. (2016). On the sense of agency and of object permanence in robots. In *2016 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)* (pp. 166–171).

Bengio, Y. & Fischer, A. (2015). Early inference in energy-based models approximates back-propagation. arXiv 1510.02777.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag.

Blakemore, S. J., Wolpert, D., & Frith, C. (2000). Why can't you tickle yourself? *Neuroreport*, 11(11), 11–16.

Blakemore, S. J., Wolpert, D. M., & Frith, C. D. (1998). Central cancellation of self-produced tickle sensation. *Nature Neuroscience*, 1(7), 635–640.

Brown, H., Friston, K., & Bestmann, S. (2011). Active Inference, attention, and motor preparation. *Frontiers in Psychology*, 2, 218.

Butz, M. V., Bilkey, D., Humaidan, D., Knott, A., & Otte, S. (2019). Learning, planning, and control in a monolithic neural event inference architecture. *Neural Networks*, 117, 135–144.

Chien, J. & Tsou, K. (2018). Convolutional Neural Turing Machine for Speech Separation. In *2018 11th International Symposium on Chinese Spoken Language Processing (ISCSLP)* (pp. 81–85).

Choi, M., Matsumoto, T., Jung, M., & Tani, J. (2018). Generating goal-directed visuomotor plans based on learning using a predictive coding type deep visuomotor recurrent neural network model. arXiv abs/1803.02578.

Collier, M. & Beel, J. (2019). Machine translation with memory augmented neural networks. In *Proceedings of Machine Translation Summit XVII, Volume 1: Research Track* (pp. 172–181).

de Brouwer, S., Missal, M., & Lefèvre, P. (2001). Role of retinal slip in the prediction of target motion during smooth and saccadic pursuit. *Journal of Neurophysiology*, 86, 550–558.

Denton, E. & Fergus, R. (2018). Stochastic video generation with a learned prior. In J. Dy & A. Krause (Eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research* (pp. 1174–1183). Stockholmsmässan, Stockholm Sweden: PMLR.

Downing, P. E. (2000). Interactions between visual working memory and selective attention. *Psychol Sci*, 11(6), 467–473.

Edelman, G. M. (1993). Neural darwinism: Selection and reentrant signaling in higher brain function. *Neuron*, 10(2), 115–125.

Eliasmith, C. (2013). *How to Build a Brain: A Neural Architecture for Biological Cognition*. Oxford University Press.

Eliasmith, C., Stewart, T. C., Choo, X., Bekolay, T., DeWolf, T., Tang, Y., & Rasmussen, D. (2012). A large-scale model of the functioning brain. *Science*, 338(6111), 1202–1205.

Epstein, M. L. (1980). The relationship of mental imagery and mental rehearsal to performance of a motor task. *Journal of Sport Psychology*, 2(3), 211–220.

Evans, G. (1982). *The Varieties of Reference*. Oxford University Press.

Faradonbeh, S. M. & Esfahani, F. S. (2019). A review on neural turing machine. arXiv abs/1904.05061.

Finn, C. & Levine, S. (2017). Deep visual foresight for planning robot motion. In *2017 IEEE International Conference on Robotics and Automation, ICRA 2017, Singapore, Singapore, May 29 - June 3, 2017* (pp. 2786–2793).: IEEE.

Finnveden, L., Jansson, Y., & Lindeberg, T. (2020). Understanding when spatial transformer networks do not support invariance, and what to do about it. arXiv cs.CV/2004.11678.

Friston, K. (2005). A theory of cortical responses. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 360(1456), 815–836.

Friston, K. (2010). The free-energy principle: A unified brain theory? *Nature Reviews Neuroscience*, 11, 127–38.

Friston, K. (2013). Active inference and free energy. *Behavioral and Brain Sciences*, 36(3), 212–213.

Friston, K., Kilner, J., & Harrison, L. (2006). A free energy principle for the brain. *Journal of Physiology-Paris*, 100(1), 70–87. Theoretical and Computational Neuroscience: Understanding Brain Functions.

Fukushima, K. & Miyake, S. (1982). Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In S.-i. Amari & M. A. Arbib (Eds.), *Competition and Cooperation in Neural Nets* (pp. 267–285). Berlin, Heidelberg: Springer Berlin Heidelberg.

Fuster, J. M. (2004). Upper processing stages of the perception–action cycle. *Trends in Cognitive Sciences*, 8(4), 143–145.

Fuster, J. M. (2015). *The Prefrontal Cortex*. Elsevier Science.

Fuster, J. M. & Jervey, J. P. (1982). Neuronal firing in the inferotemporal cortex of the monkey in a visual memory task. *Journal of Neuroscience*, 2(3), 361–375.

Goldman-Rakic, P. (1995). Cellular basis of working memory. *Neuron*, 14(3), 477–485.

Goodale, M. A. & Milner, A. (1992). Separate visual pathways for perception and action. *Trends in Neurosciences*, 15(1), 20–25.

Graves, A., Wayne, G., & Danihelka, I. (2014). Neural Turing Machines. arXiv abs/1410.5401.

Graves, A., Wayne, G., Reynolds, M., Harley, T., Danihelka, I., Grabska-Barwińska, A., Colmenarejo, S. G., Grefenstette, E., Ramalho, T., Agapiou, J., Badia, A. P., Hermann, K. M., Zwols, Y., Ostrovski, G., Cain, A., King, H., Summerfield, C., Blunsom, P., Kavukcuoglu, K., & Hassabis, D. (2016). Hybrid computing using a neural network with dynamic external memory. *Nature*, 538(7626), 471–476.

Gregor, K., Danihelka, I., Graves, A., & Wierstra, D. (2015). DRAW: A recurrent neural network for image generation. arXiv abs/1502.04623.

Harris, C. M. & Wolpert, D. M. (1998). Signal-dependent noise determines motor planning. *Nature*, 394(6695), 780–784.

Harrison, S. A. & Tong, F. (2009). Decoding reveals the contents of visual working memory in early visual areas. *Nature*, 458(7238), 632–635.

Hochreiter, S. & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8), 1735–1780.

Isomura, T., Shimazaki, H., & Friston, K. (2020). Canonical neural networks perform active inference. *bioRxiv*.

Ito, M. (1970). Neurophysiological aspects of the cerebellar motor control system. *International Journal of Neurology*, 7(2), 162–176.

Jaderberg, M., Simonyan, K., Zisserman, A., & kavukcuoglu, k. (2015). Spatial Transformer Networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 28* (pp. 2017–2025). Curran Associates, Inc.

Jeannerod, M. (1994). The representing brain: Neural correlates of motor intention and imagery. *Behavioral and Brain Sciences*, 17(2), 187–202.

Jung, M., Matsumoto, T., & Tani, J. (2019). Goal-directed behavior under variational predictive coding: Dynamic organization of visual attention and working memory. In *2019 IEEE/RSJ International Conference on Intelligent Robot and Systems (IROS)* (pp. 1040–1047).

Kaplan, R. & Friston, K. J. (2018). Planning and navigation as active inference. *Biological Cybernetics*, 112(4), 323–343.

Kingma, D. P. & Welling, M. (2014). Auto-encoding variational bayes. In Y. Bengio & Y. LeCun (Eds.), *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

Krichmar, J. L., Nitz, D. A., Gally, J. A., & Edelman, G. M. (2005). Characterizing functional hippocampal pathways in a brain-based device as it solves a spatial memory task. *Proceedings of the National Academy of Sciences*, 102(6), 2111–2116.

Kumar, S., Joseph, S., Gander, P. E., Barascud, N., Halpern, A. R., & Griffiths, T. D. (2016). A brain system for auditory working memory. *Journal of Neuroscience*, 36(16), 4492–4505.

Kuniyoshi, Y., Inaba, M., & Inoue, H. (1994). Learning by watching: Extracting reusable task knowledge from visual observation of human performance. *IEEE Transactions on Robotics and Automation*, 10(6), 799–822.

Lang, C., Schillaci, G., & Hafner, V. V. (2018). A deep convolutional neural network model for sense of agency and object permanence in robots. In *2018 Joint IEEE 8th International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)* (pp. 257–262).

Le, H., Tran, T., Nguyen, T., & Venkatesh, S. (2018). Variational memory encoder-decoder. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 31* (pp. 1508–1518). Curran Associates, Inc.

Li, M., Liu, J., & Tsien, J. Z. (2016). Theory of connectivity: Nature and nurture of cell assemblies and cognitive computation. *Frontiers in Neural Circuits*, 10, 34.

Luck, S. J. & Vogel, E. K. (1997). The capacity of visual working memory for features and conjunctions. *Nature*, 390(6657), 279–281.

McClelland, J. L. & Plaut, D. C. (1999). Does generalization in infant learning implicate abstract algebra-like rules? *Trends in Cognitive Sciences (Regular Edition)*, 3(5), 166–168.

Mesulam, M. M. (1998). From sensation to cognition. *Brain*, 121(6), 1013–1052.

Miall, R. C., Weir, D. J., Wolpert, D. M., & Stein, J. F. (1993). Is the cerebellum a smith predictor? *Journal of Motor Behavior*, 25(3), 203–216.

Murata, S., Namikawa, J., Arie, H., Sugano, S., & Tani, J. (2013). Learning to reproduce fluctuating time series by inferring their time-dependent stochastic properties: Application in robot learning via tutoring. *IEEE Transactions on Cognitive and Developmental Systems*, 5(4), 298–310.

Murata, S., Yamashita, Y., Arie, H., Ogata, T., Sugano, S., & Tani, J. (2017). Learning to perceive the world as probabilistic or deterministic via interaction with others: A neuro-robotics experiment. *IEEE Trans Neural Netw Learn Syst*, 28(4), 830–848.

Nair, A., Pong, V., Dalal, M., Bahl, S., Lin, S., & Levine, S. (2018). Visual reinforcement learning with imagined goals. In S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada* (pp. 9209–9220).

Nyberg, L., Habib, R., McIntosh, A., & Tulving, E. (2000). Reactivation of encoding-related brain activity during memory retrieval. *Proceedings of the National Academy of Sciences of the United States of America*, 97, 11120–4.

Ohata, W. & Tani, J. (2020). Investigation of the sense of agency in social cognition, based on frameworks of predictive coding and active inference: A simulation study on multimodal imitative interaction. *Frontiers in Neurorobotics*, 14, 61.

O'Reilly, R. C. (2006). Biologically based computational models of high-level cognition. *Science (New York, N.Y.)*, 314(5796), 91–94.

O'Reilly, R. C. & Frank, M. J. (2006). Making working memory work: A computational model of learning in the prefrontal cortex and basal ganglia. *Neural Computation*, 18(2), 283–328.

Pailian, H., Störmer, V., & Alvarez, G. (2017). Neurophysiological marker of visual working memory manipulation. *Journal of Vision*, 17, 1116.

Pascanu, R., Mikolov, T., & Bengio, Y. (2013). On the difficulty of training recurrent neural networks. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings* (pp. 1310–1318).: JMLR.org.

Piaget, J. & Cook (Ed.), M. (1952). *The origins of intelligence in children*. W W Norton & Co.

Posner, M. I. (1995). *Attention in cognitive neuroscience: An overview*, (pp. 615–624). The cognitive neurosciences. The MIT Press: Cambridge, MA, US.

Rakic, P. (2009). Evolution of the neocortex: a perspective from developmental biology. *Nature Reviews. Neuroscience*, 10(10), 724–735.

Rao, R. & Ballard, D. (1999). Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, 2, 79–87.

Reddy, V. (2008). *How infants know minds*. Harvard University Press.

Rosenbaum, D. A. (1991). *Human Motor Control*. Academic Press.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1988). *Learning Representations by Back-Propagating Errors*, (pp. 696–699). MIT Press: Cambridge, MA, USA.

Saffran, J. R., Aslin, R. N., & Newport, E. L. (1996). Statistical learning by 8-month-old infants. *Science*, 274(5294), 1926–1928.

Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-k., & WOO, W.-c. (2015). Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 28* (pp. 802–810). Curran Associates, Inc.

Shima, K., Isoda, M., Mushiake, H., & Tanji, J. (2007). Categorization of behavioral sequences in the prefrontal cortex. *Nature*, 445, 315–318.

Sur, M. & Rubenstein, J. L. (2005). Patterning and plasticity of the cerebral cortex. *Science*, 310(5749), 805–810.

Tai, K. S., Bailis, P., & Valiant, G. (2019). Equivariant transformer networks. In *International Conference on Machine Learning*.

Tanaka, K. (1996). Inferotemporal cortex and object vision. *Annual Review of Neuroscience*, 19(1), 109–139. PMID: 8833438.

Tani, J. (2003). Learning to generate articulated behavior through the bottom-up and the top-down interaction processes. *Neural Networks*, 16(1), 11–23.

Tokyo Robotics Inc. (2020). Torobo Arm: Accelerate your research.

Ungerleider, L. G., Courtney, S. M., & Haxby, J. V. (1998). A neural system for human visual working memory. *Proc Natl Acad Sci U S A*, 95(3), 883–890.

Van Essen, D. C. & Maunsell, J. H. (1983). Hierarchical organization and functional streams in the visual cortex. *Trends in Neurosciences*, 6, 370–375.

Vogel, E. K. & Machizawa, M. G. (2004). Neural activity predicts individual differences in visual working memory capacity. *Nature*, 428(6984), 748–751.

Weng, J. J., Ahuja, N., & Huang, T. S. (1993). Learning recognition and segmentation of 3-d objects from 2-d images. In *Fourth International Conference on Computer Vision, ICCV 1993, Berlin, Germany, 11-14 May, 1993, Proceedings* (pp. 121–128).: IEEE Computer Society.

Werbos, P. J. (1990). Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10), 1550–1560.

Wersing, H., Kirstein, S., Götting, M., Brandl, H., Dunn, M., Mikhailova, I., Goerick, C., Steil, J., Ritter, H., & Körner, E. (2007). Online learning of objects in a biologically motivated visual architecture. *International Journal of Neural Systems*, 17, 219–30.

Wersing, H., Steil, J. J., & Ritter, H. (1997). A layered recurrent neural network for feature grouping. In W. Gerstner, A. Germond, M. Hasler, & J.-D. Nicoud (Eds.), *Artificial Neural Networks — ICANN'97* (pp. 439–444). Berlin, Heidelberg: Springer Berlin Heidelberg.

Wilson, F. A., Scalaidhe, S. P., & Goldman-Rakic, P. S. (1993). Dissociation of object and spatial processing domains in primate prefrontal cortex. *Science*, 260(5116), 1955–1958.

Wolpert, D. M. & Miall, R. C. (1996). Forward models for physiological motor control. *Neural Networks*, 9(8), 1265–1279.

Wolpert, D. M., Miall, R. C., & Kawato, M. (1998). Internal models in the cerebellum. *Trends in Cognitive Sciences (Regular Edition)*, 2(9), 338–347.

Yamashita, Y. & Jun, T. (2008). Emergence of functional hierarchy in a multiple timescale neural network model: A humanoid robot experiment. *PLOS Computational Biology*, 4(11), 1–18.