Ciljno-usmjerena vizualna pažnja koristeći duboki model aktivnog zaključivanja

Mišić, Tin

UNIVERSITY OF ZAGREB
FACULTY OF ELECTRICAL ENGINEERING AND COMPUTING

MASTER THESIS No. 513

GOAL-DIRECTED VISUAL ATTENTION USING A DEEP ACTIVE INFERENCE MODEL

Tin Mišić

Zagreb, June 2024

UNIVERSITY OF ZAGREB
FACULTY OF ELECTRICAL ENGINEERING AND COMPUTING

MASTER THESIS No. 513

GOAL-DIRECTED VISUAL ATTENTION USING A DEEP ACTIVE INFERENCE MODEL

Tin Mišić

Zagreb, June 2024

UNIVERSITY OF ZAGREB
FACULTY OF ELECTRICAL ENGINEERING AND COMPUTING

Zagreb, 04 March 2024

MASTER THESIS ASSIGNMENT No. 513

Student:	**Tin Mišić (0036521636)**
Study:	Computing
Profile:	Computer Science
Mentor:	prof. Ivan Marković

Title: **Goal-directed visual attention using a deep active inference model**

Description:

Visual attention is a complex cognitive process that depends on the current internal state of the agent and the current state of the environment in which it is located. The agent needs to actively select a part of the sensory space to focus on in order to satisfy internal needs. The aim of this thesis is to develop a deep model that will, based on visual and proprioceptive sensory input, direct attention to objects in the environment and thus satisfy its complex goals. The development of such a model relies on the free-energy principle, within the frameworks of predictive coding for predicting sensory input and active inference for the active selection of the sensory input. The developed model will be trained and tested in simulated and real conditions. The result of the work will provide insights into the ability of a deep model based on active inference for selective focus on a part of the sensory space that best fulfills the current goal. The potential of the model for integrating multiple sensory modalities to achieve attention will also be explored.

Submission date: 28 June 2024

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Zagreb, 4. ožujka 2024.

DIPLOMSKI ZADATAK br. 513

Pristupnik:	**Tin Mišić (0036521636)**
Studij:	Računarstvo
Profil:	Računarska znanost
Mentor:	prof. dr. sc. Ivan Marković

Zadatak: **Ciljno-usmjerena vizualna pažnja koristeći duboki model aktivnog zaključivanja**

Opis zadatka:

Vizualna pažnja složen je kognitivni proces koji ovisi o trenutačnom unutarnjem stanju agenta i trenutačnom stanju okoline u kojoj se on nalazi. Agent aktivno treba odabrati dio senzorskog prostora na koji treba obratiti pažnju kako bi zadovoljio unutarnje potrebe. Cilj diplomskog rada je oblikovanje dubokog modela koji će na osnovu vizualnog i propriocepcijskog senzorskog ulaza usmjeriti pažnju na objekte u okolini i time zadovoljiti svoje kompleksne ciljeve. Oblikovanje ovakvog modela oslanja se na princip slobodne energije, u okviru prediktivnog kodiranja za predviđanje senzorskih podataka i aktivnog zaključivanja za aktivan odabir senzorskih opažanja. Razvijeni model će biti učen i ispitan u simuliranim i stvarnim uvjetima. Rezultat rada pružit će uvid u sposobnost dubokog modela zasnovanog na aktivnom zaključivanju za selektivni fokus na dio senzorskog prostora koji najviše ispunjava trenutni cilj. Bit će po mogućnosti istražen i potencijal modela za integraciju više senzorskih modaliteta za ostvarenje pažnje.

Rok za predaju rada: 28. lipnja 2024.

I am deeply grateful to everyone who supported and guided me throughout the journey of completing this thesis.

Contents

1 Introduction

Attention is a critical cognitive process that significantly influences other behavioral and cognitive functions. Understanding the complex mechanisms that comprise attention is essential for uncovering how these mechanisms change throughout development and aging. Attention is notably affected in various neurological and developmental disorders. It is therefore imperative to deepen our understanding of attention to provide better support and interventions for affected individuals.

Attention is also a crucial mechanism in artificial intelligence and robotics. When faced with multiple sensory inputs (visual, auditory, tactile, etc.), it is essential for agents to selectively focus on a subset of these inputs to minimize processing load and enhance speed and performance. By prioritizing relevant information and filtering out distractions, attention mechanisms enable agents to operate more efficiently and effectively in dynamic environments. This selective focus is vital for tasks such as object recognition, navigation, and human-robot interaction, where timely and accurate responses are critical.

The goal of this thesis was to develop an active inference model of visual attention, which is directed by its sensory input to focus on simple objects in the environment. The model also needed to be able to shift attention from one object to another, depending on needs sensed from the environment. This thesis begins with a theoretical background of the free-energy principle, predictive coding and active inference in Chapter 2. Chapter 3 explains the methods used in the development of the model, giving an overview of the key parts. The implementation of the developed model is explained in Chapter 4, after which we present the results of multiple trials examining the behavior of the implemented model in Chapter 5. Finally, Chapter 6 discusses the different behaviors of the model, alongside the effects certain variables have and the model's shortcomings.

2 Theoretical Background

2.1 The Bayesian Brain Hypothesis

Biological agents, including humans, must deal with sensory uncertainty while existing in the world. This uncertain sensory information is generated by hidden states of the environment in which the agent exists and acts, and these states are unknown to the agent [1]. The Bayesian brain hypothesis proposes that biological agents possess an internal model of the world. This model maintains internal beliefs about the causes of sensory inputs and their relationships, which correspond to the hidden states of the world [2, 3]. Since these causes in the world are hierarchical in nature, the brain adopts a hierarchical model as well, with causes at multiple levels of the hierarchy. Through this generative model, the top-most beliefs about the causes influence the causes at lower levels, propagating downward and ultimately generating sensory predictions[4]. According to the Bayesian brain hypothesis, the human brain constantly predicts expected sensory input based on its internal prior beliefs and compares these predictions with the actual sensory input. When there is a mismatch, or prediction error, the brain updates its beliefs to minimize future errors, thereby continuously refining its model of the world.

Under this hypothesis, the brain does not passively receive sensory information to calculate the next best action, but actively generates perceptual predictions based on prior knowledge and current sensory input. These perceptual predictions might differ from the actual state of the world, explaining phenomena such as optical illusions and hallucinations, where the brain's predictions can override actual sensory inputs [5]. This means that, under normal neurotypical functioning, the brain optimally weighs the contributions of prior beliefs and current sensory input. For example, in a dark room, we reduce our reliance on visual input and instead depend more on our internal prior beliefs about the room's layout and other sensory modalities, such as touch. In this way,

the brain operates in an approximately Bayes-optimal manner [1, 2].

However, the hypothesis has been criticised for its unclear definition[6]. Some authors refer to it in an "as if" nature: the brain behaves as if it has a generative model and uses Bayes' theorem, without literally implementing these. Others refer to it in a "realist" sense: the brain actually has the generative model and applies Bayes' theorem, despite there being no direct evidence that this is the case. This thesis merely considers it a theoretical possibility.

2.1.1 Predictive Coding

As mentioned previously, the internal beliefs of the brain are updated based on the prediction errors that occur when there is a mismatch between the predictions of the generative model and the actual sensory input. This belief update is carried out by predictive coding, through bottom-up and top-down message passing [7, 8]. Lower parts of the hierarchical model predict sensory information, while higher parts predict the causes of those sensations. Prediction errors are calculated at each level and sent from lower levels upwards (bottom-up) to inform higher levels about discrepancies. Simultaneously, predictions from higher levels are sent downwards (top-down) to refine processing at lower levels. This bidirectional flow of information ensures that the brain continuously updates and refines its internal model, leading to more accurate and efficient sensory processing.

By minimizing prediction errors at each level of the hierarchy, the model learns the hierarchical causal structure of the world [9]. This continuous process of error minimization allows the brain to adapt to new experiences and learn over time. Predictive coding explains how perception is an active process, where the brain constantly anticipates sensory input and adjusts its predictions to align with reality. Instead of encoding and passing sensory input upwards, the brain focuses on prediction errors, enabling it to prioritize areas with the highest discrepancies. This mechanism of focusing on prediction errors implements bottom-up attention, directing cognitive resources to unexpected or significant stimuli [10]. Furthermore, inaccuracies in predictive coding may explain certain developmental and neurological disorders, such as schizophrenia, autism, anxiety disorders, and PTSD. In these conditions, the brain's ability to accurately weigh prior

beliefs against sensory input may be impaired, leading to distorted perceptions and maladaptive behaviors.

2.1.2 Variational Bayes

The brain's hierarchical model is not static and evolves with experience. The mechanism of this evolution through experience, or learning, is explained with Bayes' theorem. If latent variables $\boldsymbol{z}$ represent the agent's belief about its internal states and the states in the world, then the agent can update its beliefs using the prior distribution of those latent states and the likelihood of the sensory data $\boldsymbol{s}$. The posterior beliefs about the world $p(\boldsymbol{z}|\boldsymbol{s})$ are updated with regards to the product of prior beliefs $p(\boldsymbol{z})$ and the likelihood of the observed sensory input $p(\boldsymbol{s}|\boldsymbol{z})$ as follows:

$$p(\boldsymbol{z}|\boldsymbol{s}) = \frac{p(\boldsymbol{s}|\boldsymbol{z})p(\boldsymbol{z})}{p(\boldsymbol{s})} = \frac{p(\boldsymbol{s},\boldsymbol{z})}{p(\boldsymbol{s})} \tag{2.1}$$

Because the calculation of the marginal distribution $p(\boldsymbol{s}) = \int p(\boldsymbol{s},\boldsymbol{z})\, d\boldsymbol{z}$ is intractable because of the integration over the joint density $p(\boldsymbol{s},\boldsymbol{z})$, the posterior is approximated with a recognition distribution $q(\boldsymbol{z}) \approx p(\boldsymbol{z}|\boldsymbol{s})$. This recognition distribution is the product of inverting the generative model, in other words the inference of the causes of sensory data [2]. This approximation is done through the minimization of the Kullback-Leibler (KL) divergence between the two distributions:

$$D_{KL}[q(\boldsymbol{z})||p(\boldsymbol{z}|\boldsymbol{s})] = \int_{\boldsymbol{z}} q(\boldsymbol{z}) \ln \frac{q(\boldsymbol{z})}{p(\boldsymbol{z}|\boldsymbol{s})}\, d\boldsymbol{z} \tag{2.2}$$

Which, with equation 2.1, can be rewritten as:

$$\begin{aligned} D_{KL}[q(\boldsymbol{z})||p(\boldsymbol{z}|\boldsymbol{s})] &= \int_{\boldsymbol{z}} q(\boldsymbol{z}) \ln \frac{q(\boldsymbol{z})p(\boldsymbol{s})}{p(\boldsymbol{s},\boldsymbol{z})}\, d\boldsymbol{z} \\ &= \ln p(\boldsymbol{s}) - \int_{\boldsymbol{z}} q(\boldsymbol{z}) \ln \frac{p(\boldsymbol{s},\boldsymbol{z})}{q(\boldsymbol{z})}\, d\boldsymbol{z} \\ &= \ln p(\boldsymbol{s}) - \mathcal{L}(q) \end{aligned} \tag{2.3}$$

Where $\mathcal{L}(q)$ is the *evidence lower bound*, or *ELBO*. It serves as a lower bound on the

log evidence $\ln p(\boldsymbol{s})$ because the KL divergence is always nonnegative, therefore $\mathcal{L}(q) \leq \ln p(\boldsymbol{s})$. So, to achieve the best approximation of $p(\boldsymbol{z}|\boldsymbol{s})$ we need to minimize the KL divergence, which is equivalent to maximizing the *ELBO*.

2.2 The Free-Energy Principle

According to the free-energy principle, systems adapt and act in a way that minimizes their free-energy [4]. Free-energy is a term borrowed from physics, statistics and information theory that bounds the surprise on a sample of data, given a generative model. This principle explains how biological systems resist the natural tendency to disorder, and how it influences action and perception [2]. One of the definitions for free-energy is energy minus entropy:

$$F(\boldsymbol{z}, \boldsymbol{s}) = -\left\langle \ln p(\boldsymbol{s}, \boldsymbol{z}) \right\rangle_q + \left\langle \ln q(\boldsymbol{z}) \right\rangle_q \tag{2.4}$$

Where $\langle \cdot \rangle_q$ indicates expectation under density q. This definition shows that free-energy can be evaluated by an agent if it has a probabilistic generative model, which is expressed in terms of a likelihood and prior: $p(\boldsymbol{s}, \boldsymbol{z}) = p(\boldsymbol{s}|\boldsymbol{z})p(\boldsymbol{z})$ [11]. If this definition were to be reformulated, it shows that free-energy is equivalent to the negative *ELBO*:

$$\begin{aligned} F(\boldsymbol{z}, \boldsymbol{s}) &= -\int_z q(\boldsymbol{z}) \ln p(\boldsymbol{s}, \boldsymbol{z})\, d\boldsymbol{z} + \int_z q(\boldsymbol{z}) \ln q(\boldsymbol{z})\, d\boldsymbol{z} \\ &= \int_z q(\boldsymbol{z}) \ln \frac{q(\boldsymbol{z})}{p(\boldsymbol{s}, \boldsymbol{z})}\, d\boldsymbol{z} \\ &= -\mathcal{L}(q) \\ &= D_{KL}[q(\boldsymbol{z}) || p(\boldsymbol{z}|\boldsymbol{s})] - \ln p(\boldsymbol{s}) \end{aligned} \tag{2.5}$$

Here, minimizing the free-energy is equivalent to maximizing the *ELBO*: the KL divergence between the recognition density $q(\boldsymbol{z})$ and posterior density $p(\boldsymbol{z}|\boldsymbol{s})$ is reduced. The negative log probability of an outcome (here, $-\ln p(\boldsymbol{s})$) is defined as *surprise*. Free-energy is the upper bound on surprise, so minimizing it minimizes the surprise of a certain sampling of sensory input. Under variational Bayes, this is done by changing the internal states $\boldsymbol{z}$ of the model [11]. This minimization is far simpler to perform by the

system than the minimization of the KL divergence, as it only relies on the approximate posterior and the generative model of the system.

Under the free-energy principle, all of the variables of a system will change so that free-energy is minimized [4]. A biological system must constantly keep it self in a set of states in which it is kept alive and in which it can oppose disorder [3], and this means that surprise about its states, or the upper bound of that surprise, must be minimal. One way this is accomplished is by *perceptual inference*, where the most plausible hidden states $\boldsymbol{z}$, are those which minimize free-energy:

$$\boldsymbol{z} = \arg\min_{z} F(\boldsymbol{z}, \boldsymbol{s}) \tag{2.6}$$

With the assumptions [12] that:

- under the mean-field approximation, the recognition density can be partitioned into independent distributions: $q(\boldsymbol{z}) = \prod_i q(\boldsymbol{z}_i)$, and that
- each of these partitions is Gaussian: $q(\boldsymbol{z}_i) = \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Pi}_i^{-1})$, under the Laplace approximation,

systems can approximate the environment, which is defined as a dynamical system with additive random effects[11]:

$$\begin{aligned} \boldsymbol{s} &= \boldsymbol{g}(\boldsymbol{z}) + \boldsymbol{w}_s \\ \dot{\boldsymbol{z}} &= \boldsymbol{f}(\boldsymbol{z}) + \boldsymbol{w}_z \end{aligned} \tag{2.7}$$

with their own model:

$$\begin{aligned} \boldsymbol{s} &= g(\boldsymbol{z}) + \boldsymbol{w}_s \\ \dot{\boldsymbol{z}} &= f(\boldsymbol{z}) + \boldsymbol{w}_z \end{aligned} \tag{2.8}$$

Because of the assumption that each partition of the recognition density is Gaussian, free-energy does not depend on the hidden states $\boldsymbol{z}$, but on their most probable hypotheses, *beliefs* $\boldsymbol{\mu}$, and their precision matrices $\boldsymbol{\Pi}_i$. Now, free-energy becomes:

$$F(\boldsymbol{\mu}, \boldsymbol{s}) = -\ln p(\boldsymbol{s}, \boldsymbol{\mu}) + C = -\ln p(\boldsymbol{s}|\boldsymbol{\mu}) - \ln p(\boldsymbol{\mu}) + C \tag{2.9}$$

where C is a constant. Then, the internal model of the environment becomes:

$$\begin{aligned} \tilde{\boldsymbol{s}} &= \tilde{g}(\tilde{\boldsymbol{\mu}}) + \boldsymbol{w}_s \\ D\tilde{\boldsymbol{\mu}} &= \tilde{f}(\tilde{\boldsymbol{\mu}}) + \boldsymbol{w}_\mu \end{aligned} \tag{2.10}$$

Here, $\tilde{\boldsymbol{\mu}}$ indicates generalized coordinates of beliefs with multiple temporal orders, $\tilde{\boldsymbol{\mu}} = \{\boldsymbol{\mu}, \boldsymbol{\mu}', \boldsymbol{\mu}'', \cdots\}$, which allow for finer approximation of the environment. $D\cdot$ is the differential shift operator: $D\tilde{\boldsymbol{\mu}} = \{\boldsymbol{\mu}', \boldsymbol{\mu}'', \cdots\}$, in the generalized equation of system dynamics $\tilde{f}(\tilde{\boldsymbol{\mu}})$.

In this way, perception optimizes predictions of the generative model [2]. Once we learn what the causes to our sensations are, we cannot be surprised when what we expect actually happens.

Belief update

With the generalized coordinates of sensory data and beliefs, the likelihood and prior distributions in Eq. 2.9 also become generalized and can be partitioned within and across temporal orders d, respectively:

$$\begin{aligned} p(\tilde{\boldsymbol{s}}|\tilde{\boldsymbol{\mu}}) &= \prod_d p(\boldsymbol{s}^{[d]}|\boldsymbol{\mu}^{[d]}) \\ p(\tilde{\boldsymbol{\mu}}) &= \prod_d p(\boldsymbol{\mu}^{[d+1]}|\boldsymbol{\mu}^{[d]}) \end{aligned} \tag{2.11}$$

These partitions are assumed to be Gaussian, as mentioned previously:

$$\begin{aligned} p(\boldsymbol{s}^{[d]}|\boldsymbol{\mu}^{[d]}) &= \frac{\boldsymbol{\Pi}_{\boldsymbol{s}}}{\sqrt{(2\pi)^L}} \exp\left(-\frac{1}{2}{\boldsymbol{e}_s^{[d]}}^T \boldsymbol{\Pi}_{\boldsymbol{s}} \boldsymbol{e}_s^{[d]}\right) \\ p(\boldsymbol{\mu}^{[d+1]}|\boldsymbol{\mu}^{[d]}) &= \frac{\boldsymbol{\Pi}_{\boldsymbol{\mu}}}{\sqrt{(2\pi)^M}} \exp\left(-\frac{1}{2}{\boldsymbol{e}_\mu^{[d]}}^T \boldsymbol{\Pi}_{\boldsymbol{\mu}} \boldsymbol{e}_\mu^{[d]}\right) \end{aligned} \tag{2.12}$$

where L and M are the respective dimensions of sensory data and internal beliefs.

In these variational probability distributions, $\boldsymbol{e}_s^{[d]}$ and $\boldsymbol{e}_\mu^{[d]}$ are sensory and system dynamics prediction errors:

$$\boldsymbol{e}_s^{[d]} = \boldsymbol{s}^{[d]} - g^{[d]}(\boldsymbol{\mu}^{[d]}) = \boldsymbol{s}^{[d]} - \boldsymbol{p}^{[d]} \tag{2.13}$$

$$\boldsymbol{e}_\mu^{[d]} = \boldsymbol{\mu}^{[d+1]} - f^{[d]}(\boldsymbol{\mu}^{[d]}) \tag{2.14}$$

Because the Gaussians are smooth and differentiable, gradients can be easily computed and gradient descent performed tractably over the generalized coordinates by changing the belief $\tilde{\boldsymbol{\mu}}$ at every temporal order:

$$\dot{\tilde{\boldsymbol{\mu}}} - D\tilde{\boldsymbol{\mu}} = -\partial_{\tilde{\mu}} F(\tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{s}}) \tag{2.15}$$

Now, with Eq. 2.9, the belief update becomes:

$$\dot{\tilde{\boldsymbol{\mu}}} = D\tilde{\boldsymbol{\mu}} + \frac{\partial \tilde{g}}{\partial \tilde{\boldsymbol{\mu}}}^T \tilde{\boldsymbol{\Pi}}_s \tilde{\boldsymbol{e}}_s + \frac{\partial \tilde{f}}{\partial \tilde{\boldsymbol{\mu}}}^T \tilde{\boldsymbol{\Pi}}_\mu \tilde{\boldsymbol{e}}_\mu - D^T \tilde{\boldsymbol{\Pi}}_\mu \tilde{\boldsymbol{e}}_\mu \tag{2.16}$$

The three terms that comprise the belief update are:

- $\frac{\partial \tilde{g}}{\partial \tilde{\boldsymbol{\mu}}}^T \tilde{\boldsymbol{\Pi}}_s \tilde{\boldsymbol{e}}_s$: likelihood error computed at the sensory level, representing the free-energy gradient of the likelihood relative to the belief $\tilde{\boldsymbol{\mu}}^{[d]}$, in Eq. 2.13
- $\frac{\partial \tilde{f}}{\partial \tilde{\boldsymbol{\mu}}}^T \tilde{\boldsymbol{\Pi}}_\mu \tilde{\boldsymbol{e}}_\mu$: backward error from the next temporal order, representing the free-energy gradient relative to the belief $\tilde{\boldsymbol{\mu}}^{[d+1]}$ in Eq. 2.14
- $-D^T \tilde{\boldsymbol{\Pi}}_\mu \tilde{\boldsymbol{e}}_\mu$: forward error coming from the previous temporal order, representing the free-energy gradient relative to the belief $\tilde{\boldsymbol{\mu}}^{[d]}$ in Eq. 2.14

To summarize, the predictions generated by the model are constantly compared to the sensory input, which generates prediction errors. These prediction errors consist of

sensory errors and errors across different temporal orders of the internal belief, which together update the internal belief and drive it towards a point which minimizes free-energy, and by proxy surprise.

2.3 Active Inference

Just learning about the causes of sensations is not enough. For example, learning that touching hot things causes pain is not enough to keep us from touching them. States that are compatible with survival are called attractors, and systems tend to stay around them once close, even under small perturbations [2]. These attractors have evolved through time and they influence systems' beliefs and therefore their expectations about the state of the world. To actively bring about these expectations from beliefs, systems must actively sample the sensory space in a way that minimizes surprise, or prediction errors. In this way, optimal action is inferred by minimizing free-energy:

$$\boldsymbol{a} = \arg\min_{\boldsymbol{a}} F(\boldsymbol{\mu}, \boldsymbol{s}) \tag{2.17}$$

Since motor control is only dependent on sensory information, the action update becomes:

$$\dot{\boldsymbol{a}} = -\partial_{\boldsymbol{a}} F(\tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{s}}) = -\frac{\partial F}{\partial \tilde{\boldsymbol{s}}}\frac{\partial \tilde{\boldsymbol{s}}}{\partial \boldsymbol{a}} = -\frac{\partial \tilde{\boldsymbol{s}}}{\partial \boldsymbol{a}}^{T} \tilde{\boldsymbol{\Pi}}_s \tilde{\boldsymbol{e}}_s \tag{2.18}$$

This enables the system to update belief with likelihood contributions from each sensory modality. However, it requires knowing the inverse mapping from exteroceptive sensory data to action. Nevertheless, it is through this action that systems can actively inhabit a set of states that promote the greatest chances of survival. Active inference was shown to possibly explain biological behaviours, such as saccadic and smooth pursuit eye movements [13], binocular depth estimation [14], the rubber hand illusion [15] and motor learning [16]. It has also been proposed as an alternative to reinforcement learning or control theory when optimizing behaviour [11] and successfully applied to robotics [17, 18].

2.3.1 Attention

Attention can be both goal-directed and stimulus-driven. The former is influenced by top-down selection of stimuli, meaning that it is shaped by an agent's goals, expectations, and prior knowledge, directing its focus towards specific aspects of its environment that are relevant to its current tasks or objectives. In contrast, stimulus-driven attention is governed by bottom-up stimulus properties, which capture agents' attention due to their inherent characteristics, such as sudden movement, bright colors, or loud noises, making them stand out in the sensory landscape [10, 19].

It is this bottom-up surprise that attracts attention to those parts of the sensory space that are most unexpected. When something in the environment deviates from what the system expects, it creates a form of sensory surprise that compels the system to focus on it, often as a potential sign of something important or needing immediate response[20]. This mechanism is akin to active inference[21, 22], where surprise drives the belief system towards a state that best explains the sensory information received and thereby minimizes future surprise. Attention has also been examined from a Bayesian perspective [22, 23, 24, 20, 25], examining how attention balances expected goal-directed influences and unexpected stimulus-driven influences.

The top-down goal-directed selection of stimuli is dependent on the functioning of higher areas of the brain, particularly the prefrontal cortex (PFC)[10, 26]. The PFC is crucial for executive functions such as planning, decision-making, and moderating social behavior, and it plays a significant role in controlling attention by prioritizing stimuli that align with our goals and suppressing those that are irrelevant. This dual mechanism of attention — balancing the influence of immediate sensory inputs with higher-level cognitive processes — ensures that we can effectively navigate and respond to our complex and ever-changing environment.

2.4 Flexible Intentions

In their 2023 paper[26], Priorelli and Stoianov introduced the theory of *flexible intentions* in the context of active inference. This theory posits that for an agent to adapt to a changing environment and adjust its goals accordingly, it must utilize dynamic attractors that

influence its actions in real-time. For instance, an agent might encode within its belief system the states of multiple moving objects within a scene. In tasks such as grasping or gazing, these objects represent affordances that the agent can select from, with each belief state corresponding to the successful completion of a different affordance acting as a distinct attractor. These attractors influence the system's dynamics and actions according to the current goals, each exerting varying degrees of influence. Priorelli and Stoianov suggest that this process of weighing different attractors and directing actions based on goals is managed by the posterior parietal cortex (PPC), through a mechanism of intention coding.

The agent manipulates its beliefs of the current state of itself and the world to construct representations of future states - *intentions*, which act as priors to the current belief[26]. Intentions $\boldsymbol{h}^{(k)}$ are constructed from current beliefs $\boldsymbol{\mu}$ to act as future goal states, with K different intention functions $\boldsymbol{i}^{(k)}(\boldsymbol{\mu}) \in \mathbb{R}^M$:

$$\boldsymbol{h}^{(k)} = \boldsymbol{i}^{(k)}(\boldsymbol{\mu}) \tag{2.19}$$

These intentions and their prediction errors later influence the belief update and by proxy action. Instead of simply updating the belief based on the sensory prediction error to match the current sensory state, like the likelihood error in Eq. 2.16, they push the belief toward the desired goal states.

2.5 Variational Autoencoders

Variational Autoencoders (VAEs) are a type of generative model that combines neural networks and probabilistic inference to generate new data similar to a given training set. Being a generative model, they learn the joint distribution $p(\boldsymbol{s}, \boldsymbol{z})$, and can generate new data from the approximated input distribution, given a prior distribution $p(\boldsymbol{z})$. Unlike traditional autoencoders that encode an input into a deterministic latent space and decode it back to the original input, VAEs encode the input into a probabilistic latent space. This is achieved by encoding the input as a distribution over the latent variables, typically assuming a Gaussian distribution. This distribution is called the *encoder distribution* $q(\boldsymbol{z}|\boldsymbol{s})$, which is the recognition distribution in variational Bayes:

$$q(\boldsymbol{z}|\boldsymbol{s}) = \mathcal{N}(\boldsymbol{z}|\boldsymbol{\mu}_{\phi}, \boldsymbol{\Sigma}_{\phi}) \tag{2.20}$$

The other distribution comprising VAEs is the *decoder distribution* $p(\boldsymbol{s}|\boldsymbol{z})$ which is the generative model that generates new sensory data from the latent encodings:

$$p(\boldsymbol{s}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{s}|\boldsymbol{\mu}_{\theta}, \boldsymbol{\Sigma}_{\theta}) \tag{2.21}$$

During training, VAEs minimize the difference between the input and its reconstruction, while also ensuring that the latent space approximates the desired prior distribution, usually a standard normal distribution. This dual objective is realized through the use of the Kullback-Leibler divergence and reconstruction loss (minimized through the maximization of the sensory data expected log likelihood $p(\boldsymbol{s}|\boldsymbol{z})$). These two components constitute the *ELBO*, which is maximized during training:

$$\mathcal{L}(q) = \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{s})}[\log p(\boldsymbol{s}|\boldsymbol{z})] - \beta \cdot D_{KL}(q(\boldsymbol{z}|\boldsymbol{s}) \parallel p(\boldsymbol{z})) \tag{2.22}$$

Where $\beta \geq 1$ is the regularization hyperparameter that promotes disentanglement, balancing between reconstruction of data and the restriction of the capacity of the latent channel [27, 28]. The latter enables different dimensions of the latent space to encode for specific properties of the input space (e.g. coordinates of objects in image plane), which helps with interpretability of the latent space.

3 Methods

In this section we first give an overview of the active inference model developed for this thesis. After that we cover the beliefs and intentions, going into detail of how they are maintained and updated. Later we explain how the simple attention mechanism operates and finally overview the active inference algorithm.

3.1 Model Overview

The developed active inference model can be seen in Fig. 3.1. At the center of the model are the beliefs the agent holds about its internal states and the state of the environment. These internal states hold information about the different intrinsic causes of sensations, like the proprioceptive sensations from the agent's joints and actuators, different physiological sensations like hunger, thirst etc. The states of the environment explain the causes to exteroceptive sensations, like sight, touch, sound etc. These beliefs are passed both upwards and downwards, influencing the agent's attention and goals, as well as its perception and actions.

3.1.1 Downwards Perception

The beliefs are passed down into the generative models of the different sensory modalities. In our case these are the necessitative (g_n), proprioceptive (g_p) and visual (g_v) generative models, responsible for predicting sensations of the agent's internal needs, body position and visual input respectively. The model can be easily upgraded with more sensory modalities by expanding its beliefs and adding more generative models.

These generative models create sensory predictions $\boldsymbol{p_n}$,$\boldsymbol{p_p}$ and $\boldsymbol{p_v}$ that are compared with the actual sensory input $\boldsymbol{s_n}$,$\boldsymbol{s_p}$ and $\boldsymbol{s_v}$, producing the sensory prediction errors $\boldsymbol{e_s}$

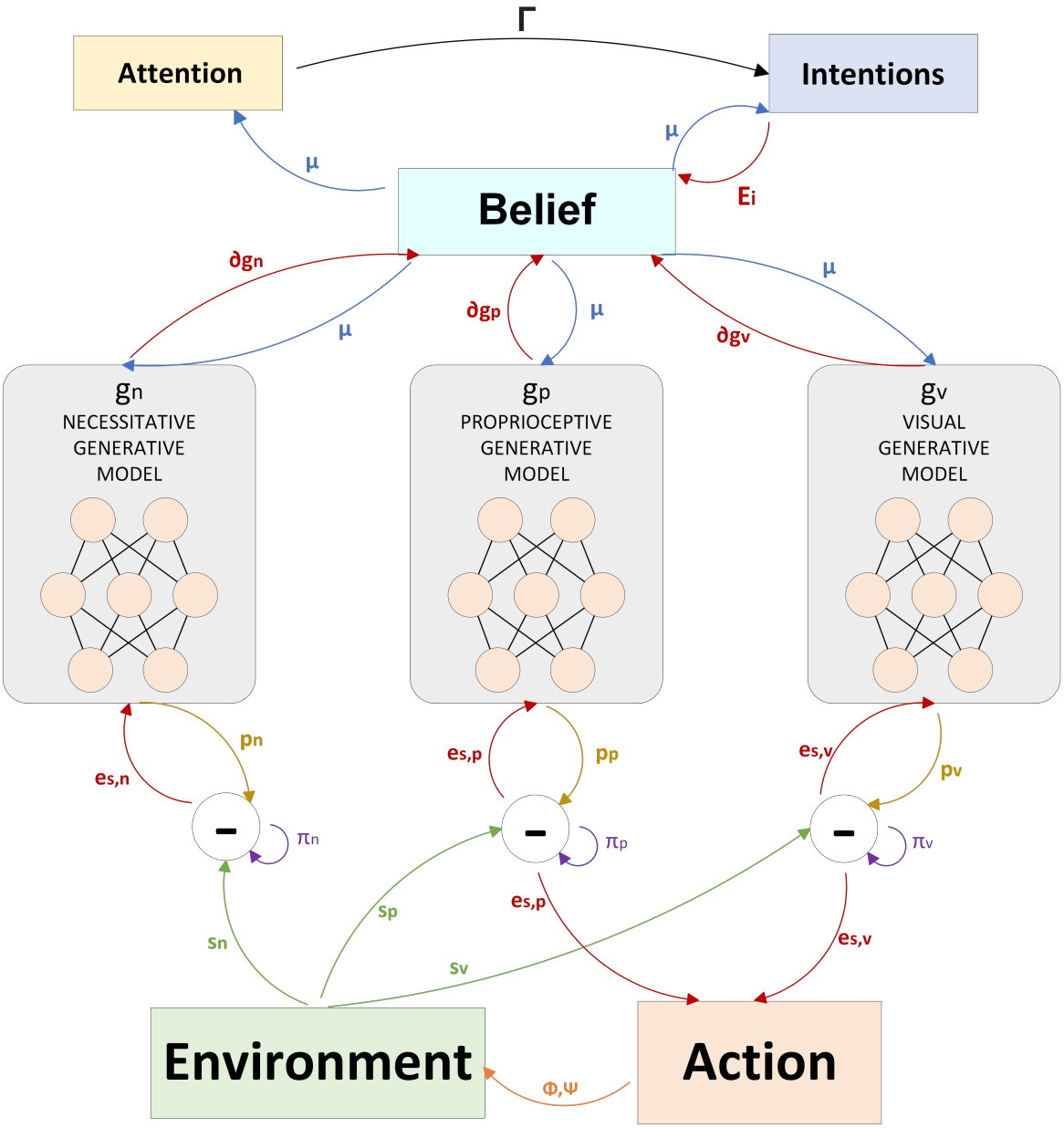

Figure 3.1: The active inference model

(Eq. 2.13). These sensory prediction errors are then either passed up to drive the belief update, or down to drive the action, through the action update shown in Eq. 2.18.

The mapping from sensory data to actions $\frac{\partial \tilde{\boldsymbol{s}}}{\partial \boldsymbol{a}}$ in the action update can be expressed through the gradients of the generative models, so the action update becomes:

$$\dot{\boldsymbol{a}} = -\frac{\partial \tilde{\boldsymbol{s}}}{\partial \boldsymbol{a}}^T \tilde{\boldsymbol{\Pi}}_s \tilde{\boldsymbol{e}}_s = -\left(\frac{\partial \tilde{g}}{\partial \tilde{\boldsymbol{\mu}}} \frac{\partial \tilde{\boldsymbol{\mu}}}{\partial \boldsymbol{a}}\right)^T \tilde{\boldsymbol{\Pi}}_s \tilde{\boldsymbol{e}}_s = -\frac{\partial \tilde{\boldsymbol{\mu}}}{\partial \boldsymbol{a}}^T \frac{\partial \tilde{g}}{\partial \tilde{\boldsymbol{\mu}}}^T \tilde{\boldsymbol{\Pi}}_s \tilde{\boldsymbol{e}}_s \tag{3.1}$$

Where $\frac{\partial \tilde{g}}{\partial \tilde{\mu}}^T \tilde{\mathbf{\Pi}}_s \tilde{\boldsymbol{e}}_s$ is the likelihood error computed at the sensory level, also used in the belief update in Eq. 2.16. The inverse model from belief to actions is:

$$\frac{\partial \tilde{\boldsymbol{\mu}}}{\partial \boldsymbol{a}} = \frac{\partial \tilde{\boldsymbol{\mu}}}{\partial q} \frac{\partial q}{\partial \boldsymbol{a}} \tag{3.2}$$

Since actions are already in the proprioceptive domain, and are defined in terms of joint velocities, the inverse proprioceptive model becomes:

$$\frac{\partial \tilde{\boldsymbol{\mu}}}{\partial \boldsymbol{a}} = \frac{\partial \tilde{\boldsymbol{\mu}}}{\partial g_p} \frac{\partial g_p}{\partial \boldsymbol{a}} \approx \Delta_t \frac{\partial g_p}{\partial \tilde{\boldsymbol{\mu}}}^{-1} \tag{3.3}$$

Where Δ_t approximates $\frac{\partial g_p}{\partial \boldsymbol{a}}$. The inverse model from visual belief to actions requires the mapping of visual belief to proprioception to be known, which becomes easier if the latent space of the visual generative model is assumed to be continuous and highly disentangled:

$$\frac{\partial \tilde{\boldsymbol{\mu}}}{\partial \boldsymbol{a}} = \frac{\partial \tilde{\boldsymbol{\mu}}}{\partial g_v} \frac{\partial g_v}{\partial \boldsymbol{a}} \tag{3.4}$$

To simplify, this inverse visual model converts the visual likelihood error into the proprioceptive domain (e.g. from pixels into angles), and helps drive action. The contribution of internal needs to action is not considered in this thesis, because the complex inverse mapping from needs to actions requires higher level mechanisms and is highly dependent on the current state of the environment. This is in a way achieved by attention.

3.1.2 Upwards Intentions and Attention

The beliefs are also passed onto the intention module that produces static and dynamic intentions or attractors that drive the belief update. This is done through the intention prediction errors $\boldsymbol{E_i}$ defined as the differences between every intention and the current belief:

$$\boldsymbol{E}^{(i)} = [(\boldsymbol{h}^{(0)} - \boldsymbol{\mu}) \cdots (\boldsymbol{h}^{(K)} - \boldsymbol{\mu})] = [\boldsymbol{e}^{i,0} \cdots \boldsymbol{e}^{i,K}] \tag{3.5}$$

These intention errors update the belief through the generalized equations of system

dynamics, with the gain k:

$$f^{(l)}(\boldsymbol{\mu}) = k\boldsymbol{e}^{i,l} + \boldsymbol{w}_{\mu}^{(l)} \tag{3.6}$$

These errors serve as attractors toward which the belief is pulled, where the pull is proportional to the error.

Attention weighs the different intentions through the intention weights vector $\boldsymbol{\Gamma}$. In this thesis, these weights are set to be mutually exclusive, that is, only one of the intentions can be active at one time, to avoid attractor conflict. $\boldsymbol{\Gamma}$ is determined from current beliefs $\boldsymbol{\mu}$, and takes into account the most adequate attractor to satisfy internal needs, while also considering the feasibility of achieving that goal - for example, how close an object is.

3.2 Belief

Since the belief needs to hold the internal states of the agent and the states of the environment, it is organized as $\boldsymbol{\mu} = [\boldsymbol{\mu}_{needs}, \boldsymbol{\mu}_{proprioceptive}, \boldsymbol{\mu}_{visual}]$. A graphical representation can be seen in Fig. 3.2.

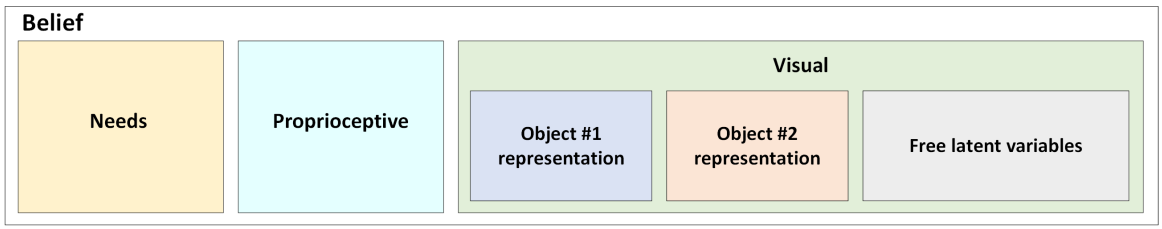

Figure 3.2: The model belief structure

The 'needs' part represents the internal physiological needs, which in our case is simplified to needs for certain types of objects. These needs act as indicators as to what object is needed, and the magnitude of that need. The proprioceptive part represents the body position of the agent, in our case the pitch and yaw angles of the camera. These are set in the global coordinate system. The visual part is the representation of the state of the environment, that is, the belief over the field of vision. In our case, it holds easily interpretable pixel coordinates of the objects in the image.

Each of these parts feeds into their respective generative model, with the visual part

having a restriction that it needs to be easily interpretable, as in Fig. 3.2. This is achieved with VAE dissentanglement, where the first n latent variables represent the attributes of the first possible object, the second n represent the attributes of the second and so on, while the remaining latent variables are free to adapt to other phenomena in the visual field (e.g. if objects overlap each other, or if one or more objects is not visible). This thesis only considers two possible objects in the environment: a red ball and a blue ball.

3.2.1 Belief Update

In this thesis, the belief has three temporal orders, $\boldsymbol{\mu}, \boldsymbol{\mu}'$ and $\boldsymbol{\mu}''$. From Eq. 2.16 we further derive the update for the individual orders[1]:

$$\begin{aligned}\dot{\tilde{\boldsymbol{\mu}}} &= D\tilde{\boldsymbol{\mu}} - \partial_{\tilde{\mu}} F(\tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{s}}) \\ &= \begin{bmatrix} \boldsymbol{\mu}' + \boldsymbol{G}^T(\boldsymbol{\pi} \odot \boldsymbol{E_s}) + (\boldsymbol{F} \odot \boldsymbol{E}^{(f)})\boldsymbol{\gamma}^T \\ \boldsymbol{\mu}'' - \boldsymbol{E}^{(f)}\boldsymbol{\gamma}^T \end{bmatrix}\end{aligned} \tag{3.7}$$

Where the prediction errors of the dynamics functions $\boldsymbol{E}^{(f)}$ and sensory prediction errors $\boldsymbol{E_s}$ are defined as:

$$\begin{aligned}\boldsymbol{E}^{(f)} &= \boldsymbol{\mu}' - \boldsymbol{E}^{(i)} \\ \boldsymbol{E_s} &= \boldsymbol{s} - \boldsymbol{p}\end{aligned} \tag{3.8}$$

Here $\boldsymbol{G}$ represents all gradients of the sensory generative models, $\boldsymbol{F}$ represents gradients of the dynamics functions and $\boldsymbol{\pi}$ and $\boldsymbol{\gamma}$ represent sensory and intention precisions, respectively. Most state-of-the-art implementations ignore the 0th order backward error as the attractor, which simplifies computation as calculating the dynamics functions gradients becomes unnecessary. The belief update can then be approximated as:

$$\dot{\tilde{\boldsymbol{\mu}}} \approx \begin{bmatrix} \boldsymbol{\mu}' + \boldsymbol{G}^T(\boldsymbol{\pi} \odot \boldsymbol{E_s}) \\ \boldsymbol{\mu}'' - \boldsymbol{E}^{(f)}\boldsymbol{\gamma}^T \end{bmatrix} = \begin{bmatrix} \boldsymbol{\mu}' + \boldsymbol{\epsilon}_s \\ \boldsymbol{\mu}'' - \boldsymbol{\epsilon}^{(i)} \end{bmatrix} \tag{3.9}$$

Where $\boldsymbol{\epsilon}_s$ and $\boldsymbol{\epsilon}^{(i)}$ are the precision-weighed sensory and prediction error contributions.

[1] The 2nd temporal order $\boldsymbol{\mu}''$ is not updated and is constant at $\mathbf{0}$

3.2.2 Precisions

Precisions play a major role in predictive coding, determining how much a certain sensory modality or intention contributes to the belief update. In the case of sensory precisions $\boldsymbol{\pi}$, each of them represents how much each sensory modality can be trusted. For example, we tend to trust our vision less in a dark room, or our sense of touch when wearing oven mitts. The precisions for the individual modalities are also regulated with the variable α, which determines the relationship between the two modalities that influence action (proprioceptive and visual). The final precisions for the modalities are then:

$$\begin{aligned} \boldsymbol{\pi}_{needs} &= \boldsymbol{\pi}'_{needs} \\ \boldsymbol{\pi}_{proprioceptive} &= \alpha\boldsymbol{\pi}'_{proprioceptive} \\ \boldsymbol{\pi}_{visual} &= (1-\alpha)\boldsymbol{\pi}'_{visual} \end{aligned} \tag{3.10}$$

The intention precisions regulate the strength with which intentions affect the belief update. They could be different for each modality part in the intentions, and different for each intention, so they are organized in a matrix $\boldsymbol{B}$. The final intention precisions $\boldsymbol{\gamma}$ for all intentions are then regulated by the intention weight vector $\boldsymbol{\Gamma}$ which selects one intention, and represses others:

$$\boldsymbol{\gamma} = \boldsymbol{\Gamma}\boldsymbol{B} \tag{3.11}$$

Another one of the precision variables is the gain k, serving as a gain to intention errors in the system dynamics functions (Eq. 3.6). How changes to these precisions affect the behavior of the system will be discussed in Chapter 6.

3.3 Intentions

Intentions drive the belief from above, generating attractors towards goal states which the agent tries to accomplish. In our case there is an intention for each object that can appear in the agents field of view. Intentions are derived from the current belief (enabling dynamic/flexible intentions) and static goal states. Since intentions are future beliefs, they have the same shape and are divided into the same parts (needs, proprioceptive and

visual) as beliefs.

The 'needs' part remains unchanged and is copied from the current belief:

$$\boldsymbol{h}^{(k)}_{needs} = \boldsymbol{\mu}_{needs} \tag{3.12}$$

This way, the 'needs' part is only updated from the sensory input. The proprioceptive part of the intentions is dynamic and depends not only on the current proprioceptive belief but on the dynamic visual part. With our visual belief being easily interpretable, we can extract the pixel coordinates of each object in the scene. If we assume that the intrinsic matrix of the camera is known, we can easily determine where the objects are located in the image in terms of relative pitch and yaw angles[2]. These relative angles are then added to the current proprioceptive belief to get proprioceptive intentions of each object, represented in the global angle positions:

$$\boldsymbol{h}^{(k)}_{proprioceptive} = \boldsymbol{\mu}_{proprioceptive} + pixels\_to\_angles(extract\_pixels(\boldsymbol{\mu}_{visual})) \tag{3.13}$$

The $extract\_pixels()$ function trivially extracts the object representations from the visual belief. However, this could be a complex function that extracts object representations from an entangled latent space, which is not explored here for simplicity.

The visual part is derived both statically and dynamically. The static goal is simply the mean latent representation of all of the training images in which the desired object is in the center of the image:

$$\boldsymbol{h}^{(k)}_{visual} = mean(\boldsymbol{c}_k) \tag{3.14}$$

Where $\boldsymbol{c}_k$ is the array of latent representations where object k is in the center. The dynamic goal is the latent representation of an image with the goal object centered, which is closest to the current belief. This pushes the goal object towards the center, while preserving the states of other objects. Assuming a continuous latent space, the visual part of the intention is calculated as the one with the smallest Euclidean distance:

[2]The angles are relative to the center of the image so that when the agent moves accordingly, the selected object will be centered.

$$\boldsymbol{h}_{visual}^{(k)} = closest(\boldsymbol{c}_k) \tag{3.15}$$

These intentions are calculated for each object and weighted with weights $\boldsymbol{\Gamma}$ from the attention module.

3.4 Algorithm

The active inference algorithm can be written as:

Algorithm 1: Active inference agent with visual attention

Require: $\boldsymbol{c}, \boldsymbol{S}, \boldsymbol{\gamma}, \boldsymbol{\pi}, \Delta_t$

$\tilde{\boldsymbol{\mu}} \leftarrow initBelief()$
while $t \leq T$ **do**
  $\boldsymbol{P} \leftarrow generatePredictions(\boldsymbol{\mu})$
  $\boldsymbol{H} \leftarrow generateIntentions(\boldsymbol{\mu}, \boldsymbol{c})$ ▷ from belief and centered representations
  $\boldsymbol{\Gamma} \leftarrow attention(\boldsymbol{\mu})$ ▷ intention weights from attention
  $\boldsymbol{E}^{(i)} \leftarrow \boldsymbol{H} - \boldsymbol{\mu}$
  $\boldsymbol{E}^{(f)} \leftarrow \boldsymbol{\mu}' - k\boldsymbol{E}^{(i)}$ ▷ dynamics prediction errors
  $\boldsymbol{E}_s \leftarrow \boldsymbol{S} - \boldsymbol{P}$ ▷ sensory prediction errors
  $\epsilon_s \leftarrow \boldsymbol{G}^T(\boldsymbol{\pi} \odot \boldsymbol{E}_s)$ ▷ weighted sensory errors
  $\epsilon^{(f)} \leftarrow \boldsymbol{E}^{(f)}\boldsymbol{\gamma}^T$ ▷ weighted intention errors
  $\dot{\boldsymbol{\mu}} \leftarrow \boldsymbol{\mu}' + \epsilon_s$ ▷ belief update
  $\dot{\boldsymbol{\mu}}' \leftarrow \boldsymbol{\mu}'' - \epsilon^{(f)}$
  $\dot{\boldsymbol{a}} \leftarrow -\partial_a \boldsymbol{\mu} \epsilon_s$ ▷ action update
  $\tilde{\boldsymbol{\mu}} \leftarrow \tilde{\boldsymbol{\mu}} + \Delta_t \dot{\tilde{\boldsymbol{\mu}}}$ ▷ update beliefs and action
  $\boldsymbol{a} \leftarrow \boldsymbol{a} + \Delta_t \dot{\boldsymbol{a}}$
end while

4 Implementation

In this section we will go over the implementation of the active inference model. The model was implemented in Python and ROS2, using Gazebo as the simulation environment. The active inference code was modified from [26] and the VAE training code from [29]. The code is open source and available at `https://github.com/TinMisic/AIF---visual-attention.git`. In this implementation the task of the active inference model is to pay attention to two balls, one red and the other blue. It must also change focus from one object to another, depending on the sensory input.

4.1 Simulation

We will first go over the world used in the simulation and give an overview of the most important ROS Nodes in the simulation. After that each of the sensory modalities will be explained as well as the nature of the sensory data used.

4.1.1 World and Nodes

The world contains a camera object located at coordinates $[0.0, 0.0, 1.0]$, and which is pointed so that it looks in the direction of the positive x axis. There are also four invisible collision walls whose function is to limit the movement of the balls along the yz plane within the ranges of $-5 < y < 5, -4 < z < 6$. Movement along the x axis is fixed at 4. The two balls are spawned into the world and given random velocity vectors in the yz plane. The simulation environment with the camera and balls can be seen in Fig. 4.1. The nodes active during the simulation can be seen in Fig. 4.2.

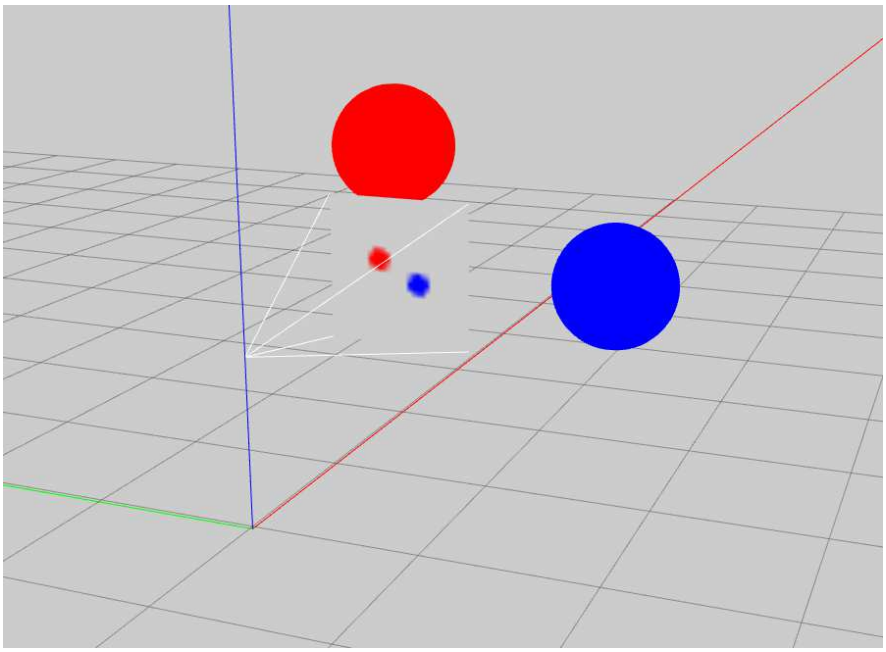

Figure 4.1: The simulation environment

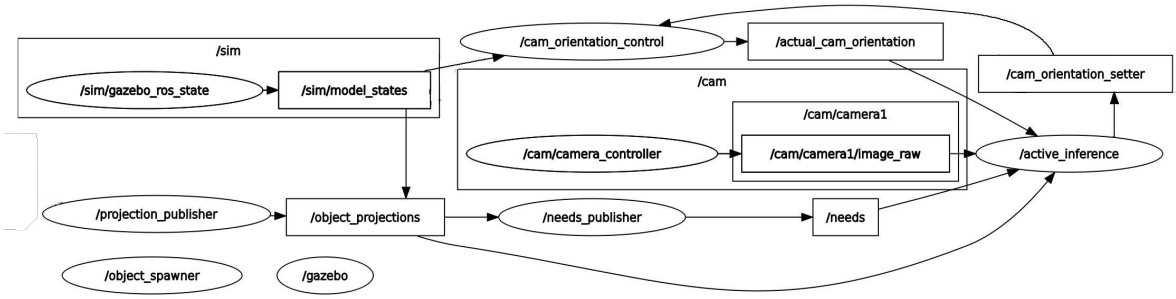

Figure 4.2: Node graph during simulation

/sim and /cam nodes

The `/sim` and `/cam` nodes are nodes crated by Gazebo and they provide model states and images, respectively. The `/model_states` topic holds messages with the information of object positions and orientations. This is important for both controlling the camera orientation and knowing the object projections needed for the `/needs_publisher`. Note that the `/active_inference` node also subscribes to the `/object_projections` topic. However, this is only for logging purposes and the object projections are not used during inference.

/cam_orientation_control node

The `/cam_orientation_control` node is tasked with controlling the camera orientation by subscribing to the `/cam_orientation_setter` topic and publishing the orientation angles to the `/actual_cam_orientation` topic, which serves as the proprioceptive input to the `/active_inference` node. The `/cam_orientation_control` node receives the wanted camera orientation and interpolates the camera movement by a constant velocity until the desired orientation is reached.

/active_inference node

The `/active_inference` node is the main computational node of the whole simulation. It is subscribed to the `/needs`, `/actual_cam_orientation` and `/cam/camera1/image_raw` topics, which serve as the needs, proprioceptive and visual sensory inputs respectively. It is also subscribed to the `/object_projections` topic, but only for logging purposes. It's only output is the action command in the form of the desired camera orientation sent to the `/cam_orientation_setter` topic.

4.1.2 Sensory Data

Needs

As previously stated, the needs array represents bodily needs that influence the agent's actions and behaviour. In our case the needs array is an array of floating point numbers with a size of 2, one floating point number for each object. The values of the needs array are limited between 0 and 1, and represent how much a specific object is desired.

The needs array can be set by a single message sent to the `/needs` topic, or generated by the `/needs_publisher` which generates values from a random sinusoidal curve, and which also takes into account the distance of the object from the center. Even if the desire for an object is high, it will be decreased if it is close to the center of the image. This acts as a kind of reward, decreasing the need for an object once at the center. The `/needs_publisher` publishes messages at a frequency of 20 Hz. Model training by reinforcement learning is possible, but is not considered in this thesis and is left for future work.

Proprioceptive input

The proprioceptive input represents the camera orientation and is in quaternion form. It is assumed that the camera only has the control over pitch and yaw angles, and those are extracted within the `/active_inference` node and are the sensory input for the model. The camera orientation is limited to $-\frac{\pi}{2} \leq \theta, \phi \leq \frac{\pi}{2}$. The orientation messages are also published at 20Hz.

Visual input

The raw image output from the camera object in Gazebo is an RGB image with the dimensions $32 \times 32 \times 3$, with a horizontal field of viw of 1.396 radians or 80°. The images are scaled between 0 and 1 and passed on to the active inference agent. Examples can be seen in Fig. 4.3. They feature a gray background with possible red or blue balls. Images are also published at a frequency of 20Hz.

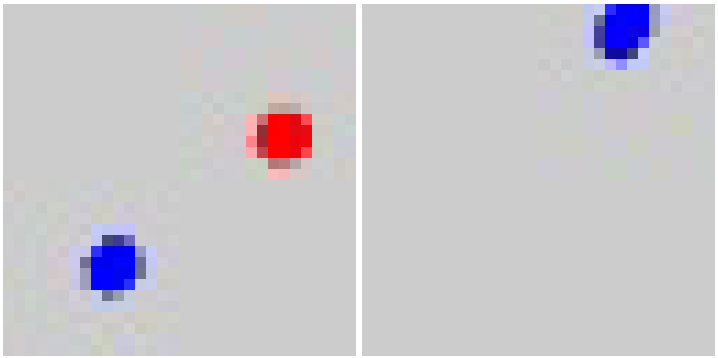

Figure 4.3: Visual input examples

Training examples for the VAE training were gathered by capturing images of the two objects scanning across the field of view, to get the best possible coverage of the scene. For each image an object projection array was saved, to aid with VAE disentanglement.

4.2 Generative Models

The generative models for the needs and proprioceptive input are trivial identity matrices, since the beliefs for those modalities are in the same space. The generative models are:

$$p_{needs} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \mu$$
$$p_{proprioceptive} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \mu \tag{4.1}$$

The last six elements of the belief belong to the latent space of the VAE. The first four elements of the latent space represent the pixel coordinates of the two objects, while the other two are free variables to accommodate invisible objects and overlapping.

4.2.1 VAE Architecture

The encoder component architecture is visible in Fig. 4.4. It's input is an image and it's outputs are the mean μ and standard variation σ, which are resampled into the sample z.

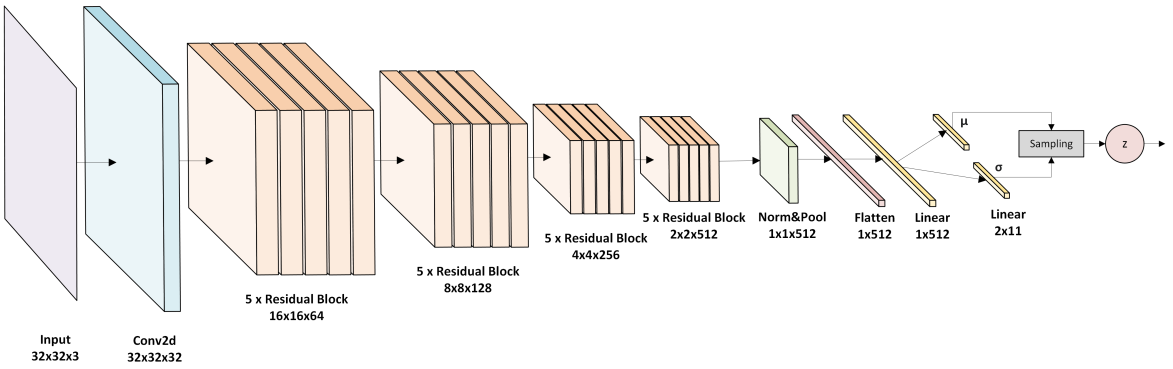

Figure 4.4: Encoder architecture

The decoder component architecture is visible in Fig. 4.5. It's input is a vector of length 10 and it's output is the visual sensory prediction. Only the decoder is used as the generative visual model during inference.

4.2.2 VAE Training

The VAE was trained in a semi-supervised manner. As mentioned previously, along with each image we have saved the object projections in that image. This enables us to force the latent space to take on a disentangled form, learning that specific latent variables code for pixel positions in the image. This is done through the KL divergence for each

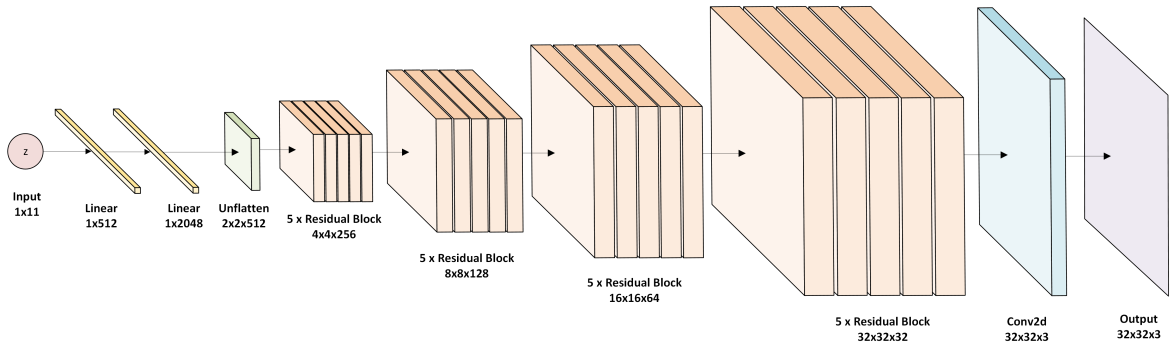

Figure 4.5: Decoder architecture

sample in the dataset. The object projections, along with the free variables set to 0, act as the prior distribution which the VAE tries to learn.

The VAE was trained on a dataset of 47,000 images and their respective projections, with a batch size of 1024 over 250 epochs. The original images and their reconstructions in the first 5 and last 5 epochs are visible in Fig. 4.6. The VAE also successfully learns to disentangle the latent space, within an average projection error of ± 2 pixels.

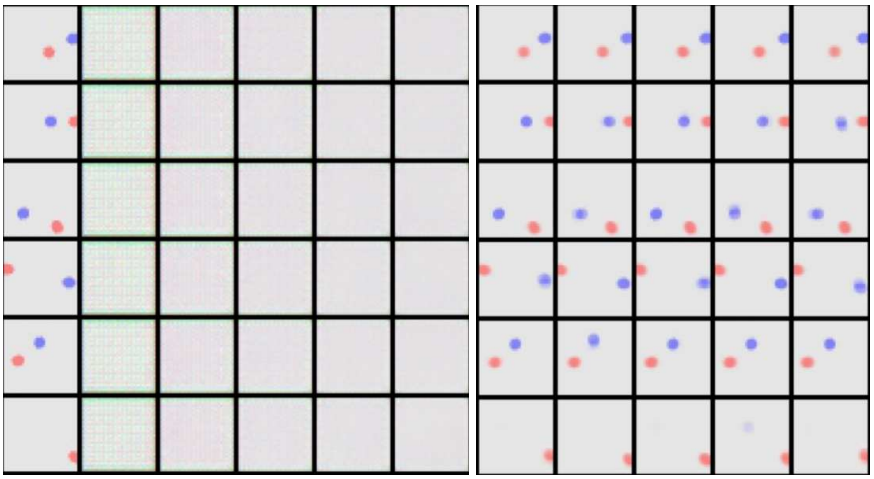

Figure 4.6: VAE reconstruction history. The first column are the original images, while the rest are reconstructions in the first 5 and last 5 epochs

4.3 Active Inference Agent

In this section we go over the most important functions within the active inference agent implementation, mainly the `/active_inference` node.

Attention module

The attention module simply checks which of the values in the needs belief is largest, and sets the according intention weights to the respective row of the $\boldsymbol{B}$ matrix, while setting all other weights to 0. The attention regulation by distance is done by proxy through the `/needs_publisher`, which will automatically lower the need if the object is at the center. This is only the case if `/needs_publisher` is running, and attention to one object can be sustained by simply publishing a constant needs message to the `/needs` topic.

Intention mode switching

How visual intentions are calculated is determined by the intention mode, which can either be `closest` or `mean`. `closest` picks the latent representation of a centered image closest to the current belief, while `mean` calculates the mean representation of all the images with the desired object in the center. These modes are changed every τ steps to utilize the advantages of each mode. How these different modes affect the behaviour of the agent will be discussed later.

POV display

During operation, the node displays the current visual sensory input, as well as the visual prediction and intention vectors, as can be seen in Fig. 4.7.

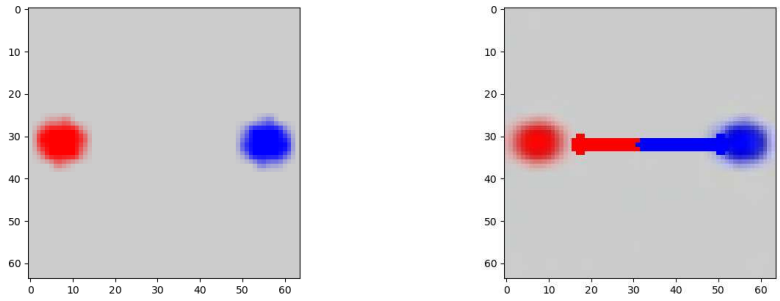

Figure 4.7: Visual sensory input and visual prediction with intention vectors for each object at the start of one trial

5 Results

In this chapter we present the results of multiple trials examining the model's performance and behaviour under different conditions. The following metrics were used to asses the model:

- L^2 **distance**: L^2 distance of object projections and estimations across simulation steps
- **Perception error of occluded object**: Perception error of object out of field of vision (FOV) across simulation steps
- **Reach error**: L^2 distance of the target object projection from the image center at the end of the trial
- **Reach time**: time until the target object projection is first centered (within 1.5 pixels from the center)
- **Reach stability**: standard deviation of L^2 distance from target reach until end of trial
- **Perception error**: L^2 distance between the true object projection and estimated target position at the end of trial
- **Perception stability**: standard deviation of distance between object projections and estimations
- **Object permanence metric**: Simple metric indicating how well an occluded ob-

ject is preserved in belief. Calculated as:

$$OPM = \frac{1}{T}\sum_{t}^{T}\frac{1}{||projection_t - estimation_t||} \tag{5.1}$$

The variable values for each trial are available in Appendix A.

5.1 Focus

In this section we examine the model's ability to choose and focus on an object. We first examine focus on static objects, assessing also the object permanence of the unfocused object once it goes out of field of vision. After that we examine focus on a moving object.

5.1.1 Static Objects

Unfocused object in FOV

Here we examine focusing on both the red and the blue ball and how the unfocused ball is preserved in belief. The starting sensory input is similar to the one in Fig. 4.7, with the distance between the two objects being 2 units in 3D space. For these trials, the needs are constant and do not change during the length of the trial.

When focusing on the red ball, the trial resulted in object projections and estimations seen in Fig. 5.1.

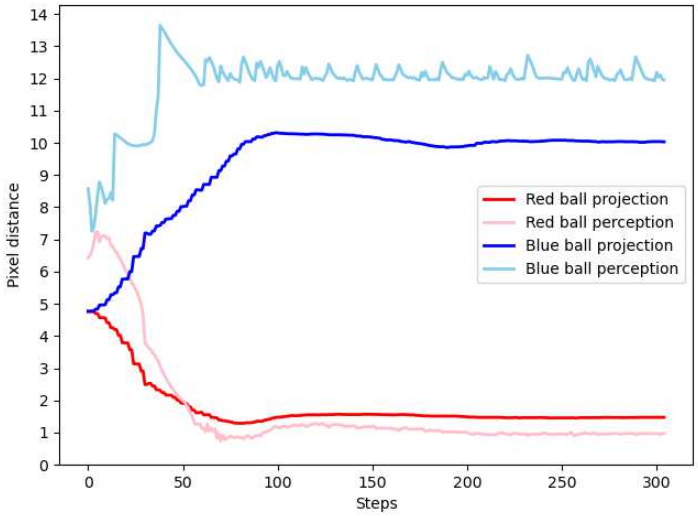

Figure 5.1: Focus on the red ball. The blue ball stays in sight upon focus on red.

We can see that the model successfully focuses on the red ball after around 100 simulation steps. It also successfully maintains perceptions of the two balls, with the blue perception being incorrect by about 2 pixels. The blue ball is still in sight, so the perception does not diverge.

Similar results are achieved when focusing on the blue ball, as seen in Fig. 5.2.

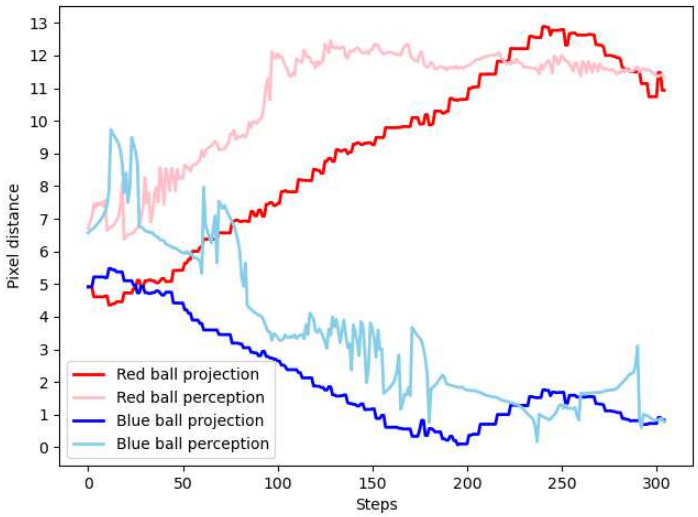

Figure 5.2: Focus on the blue ball. The red ball stays in sight upon focus on blue.

Unfocused object outside FOV

Here we examine focusing on one ball when the two are further apart (by 5 units). Upon focusing on one, the other is projected outside the FOV.

Focusing on the red ball, the projections and estimations are those in Fig. 5.3. We can see that the belief over the blue ball is somewhat maintained, even after it is out of sight (around step 50 of the trial), but it slowly diverges from the true projection. The true sensory input and perceived visual stimulus at step 50 can be seen in Fig. 5.4. Even though the blue ball is not actually visible, the model still maintains a belief over its direction. This permanence of belief is helpful when changing attention to objects out of sight.

The belief over the blue ball is almost constant for 100 steps, which can be seen more clearly in Fig. 5.5, where the difference between the perceived blue ball and its actual projection is indicated in pixels.

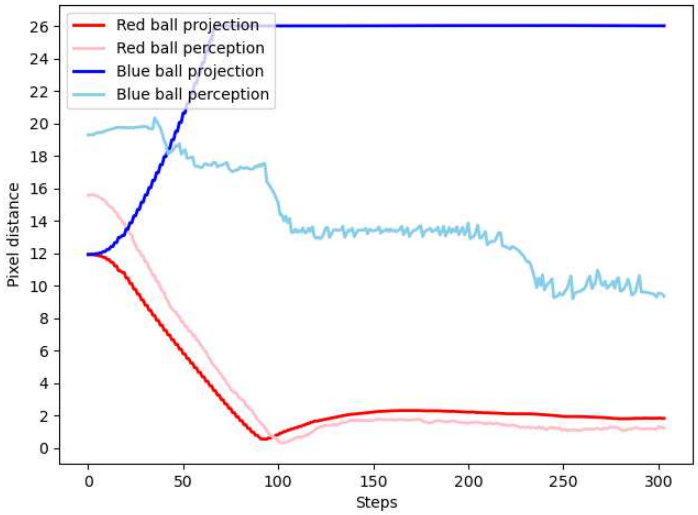

Figure 5.3: Focus on the red ball. The blue ball goes out of sight upon focus on red.

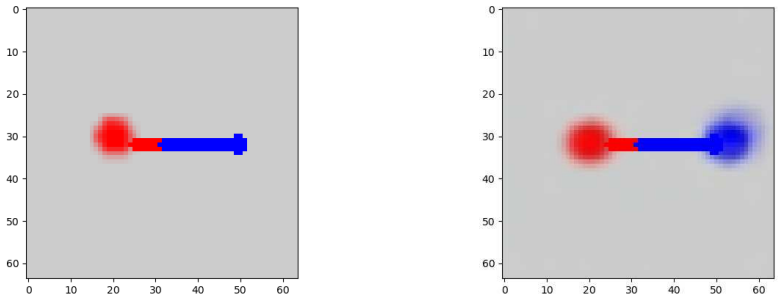

Figure 5.4: Actual sensory input and perceived sensory input at simulation step 50. Belief over the blue ball is held even after it is no longer visible.

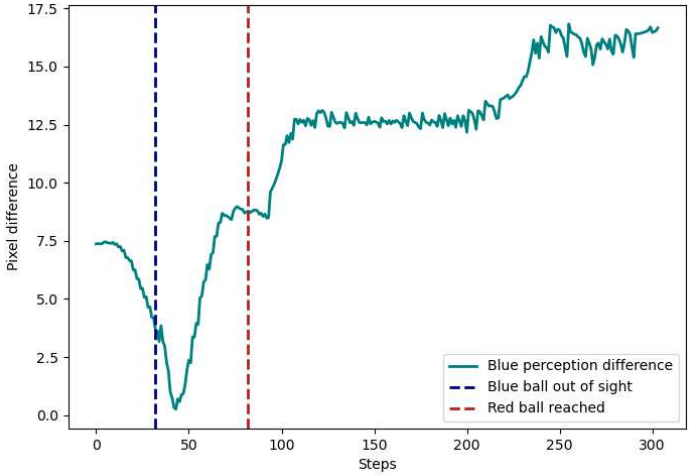

Figure 5.5: Permanence of belief over the blue balls position. Horizontal lines indicate moments in time when the blue ball is occluded and red ball is reached.

5.1.2 Moving Objects

When focusing on a moving object, the agent needs to both maintain the focused object at the center despite its movement, and maintain belief over the other moving object in the scene. The objects are moving horizontally from left to right in opposite directions. They change directions once they reach $y = \pm 1.5$ in 3D units. The projections and estimations can be seen in Fig. 5.6.

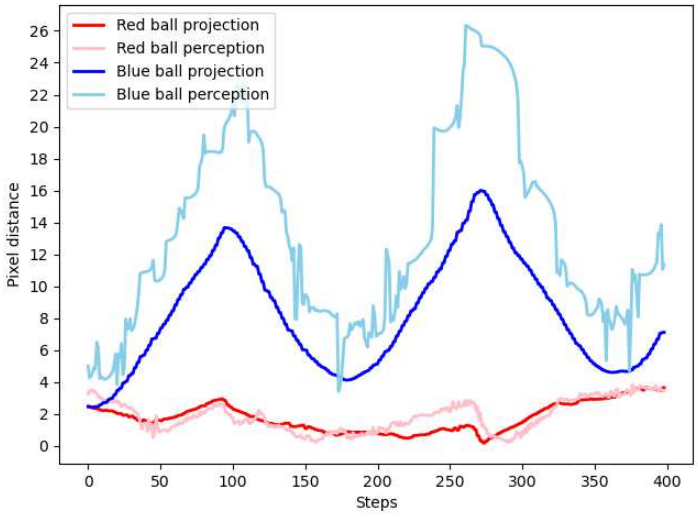

Figure 5.6: Focusing on moving red ball while maintaining belief over blue ball.

We can see that the model successfully maintains focus on the selected object and belief over the other objects position.

5.2 Attention Shift

One key ability of biological organisms is to maintain belief over many objects in the environment and to change focus between them when needed. In this section we explore the model's attention shifting ability on both static objects (in and out of FOV) and moving objects. For the sake of simplicity, the needs that guide attention are not semi-randomly generated by the `/needs_publisher`, but are changed every t simulation steps from one object to the other.

5.2.1 Static Objects in FOV

In this trial we examined the attention shifting ability when starting from focusing on red. The period of attention shift is $t = 100$ simulation steps. The projections and esti-

mations can be seen in Fig. 5.7.

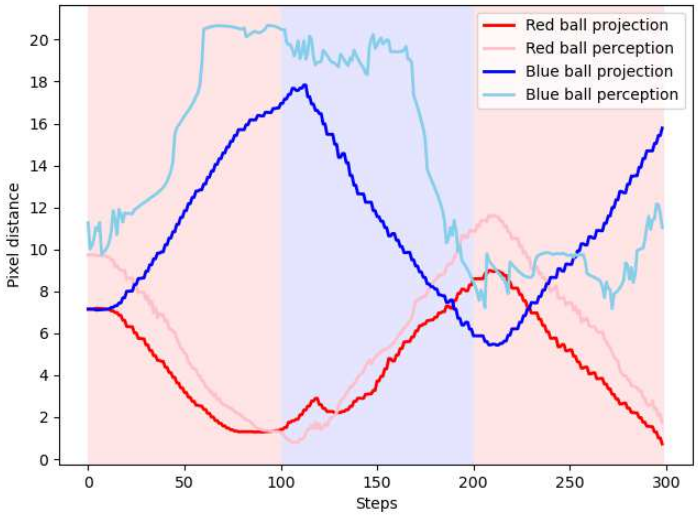

Figure 5.7: Attention shifting between static red and blue objects that are in the FOV at all times. The background color indicates the object currently being focused on.

The model successfully shifts attention from the red ball to the blue one, and then back to red once more. It successfully reaches red the first time, but it could not reach the blue one in time before the attention changed back to red.

5.2.2 Static Objects out of FOV

When the other object is out of sight, maintaining a belief over its location is crucial to be able to shift focus on it once needed. Here we examined a situation where the two objects are far enough so the unfocused object is not always in sight (5 units of distance). As before, attention is shifted every $t = 100$ steps, starting with focus on the red object. The projections and estimations are in Fig. 5.8.

We can see that the model successfully shifts attention to the blue ball, but struggles to maintain its belief, and would not manage another attention shift after the second one. The decay of the belief over the blue balls position can be seen more clearly in Fig. 5.9. The difference between the projection and estimation increases greatly after the second attention shift.

5.2.3 Moving Objects

Attention shift on moving objects is a more challenging problem, but the model fairs well with the right precision settings, as seen in Fig. 5.10. This trial's attention shift period is

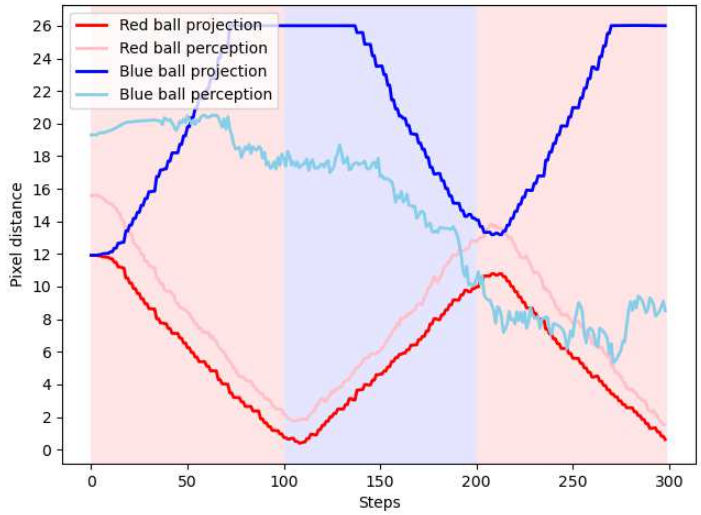

Figure 5.8: Attention shifting between static red and blue objects that are not in the FOV at all times.

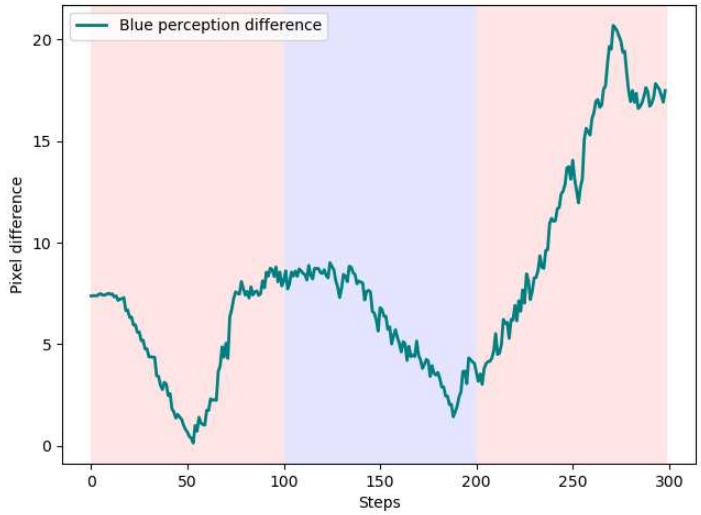

Figure 5.9: Permanence of belief over the blue balls position.

$t = 300$. Both of the objects' beliefs are well maintained even after the attention shift.

5.3 Intention mode difference

As mentioned previously, visual intentions can be calculated in two ways:

1. either by choosing the saved latent representation closest to the current belief (Eq. 3.15), or

2. by choosing the mean latent representation of the object at center (Eq. 3.14).

In this section we analyze the model's behaviour in these two modes, as well as how

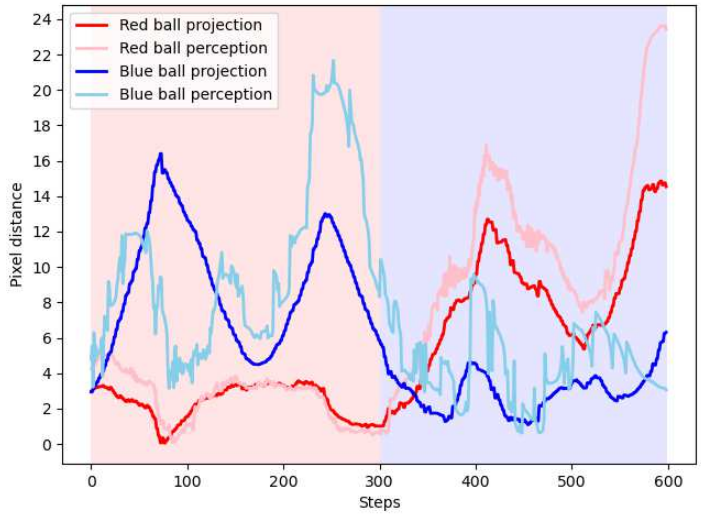

Figure 5.10: Attention shifting on moving objects during a longer trial.

it behaves when a ratio of these modes is used during inference. These differences are explored in the case of static objects, with focus on one without attention shifts.

5.3.1 Using only the `closest` mode

When using only the `closest` mode with objects constantly in FOV, we get the following results in Fig. 5.11.

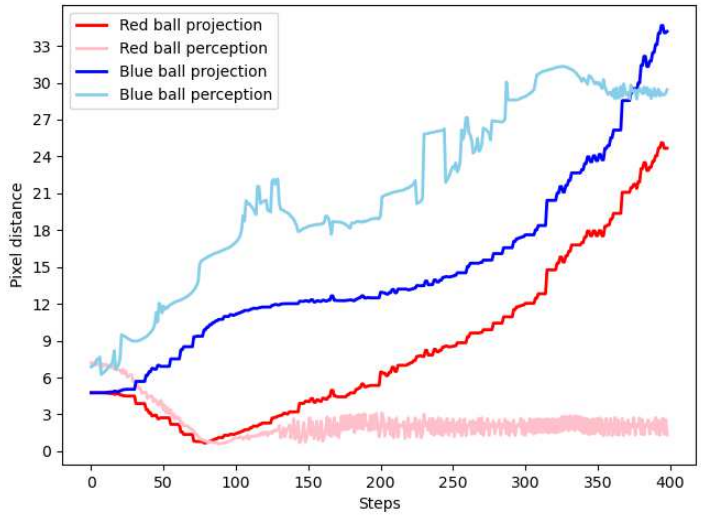

Figure 5.11: Focusing on red ball with mode `closest`, and the blue ball constantly in sight.

We can see that the model successfully focuses on the red ball, but begins to drift as its belief over the red ball is not perfectly centered. Due to already "locking in" to the red ball in its perception, it remains at the center while the true projection drifts.

When considering the case where the objects are not constantly in FOV, the model performs better. As seen in Fig. 5.12, having the unfocused object out of sight maintains its belief, while also reducing the drift over the focused red ball.

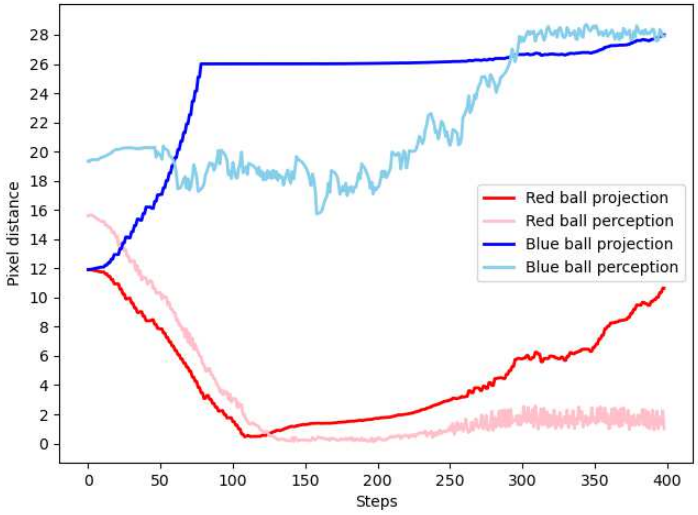

Figure 5.12: Focusing on red ball with mode `closest`, and the blue ball falls out of sight.

An advantage of the `closest` mode is that it holds beliefs over unfocused objects fairly well, while its disadvantage is that it tends to drift it's belief over the focused ball.

5.3.2 Using only the `mean` mode

When considering objects constantly in vision, the results for the `mean` mode are shown in Fig. 5.13.

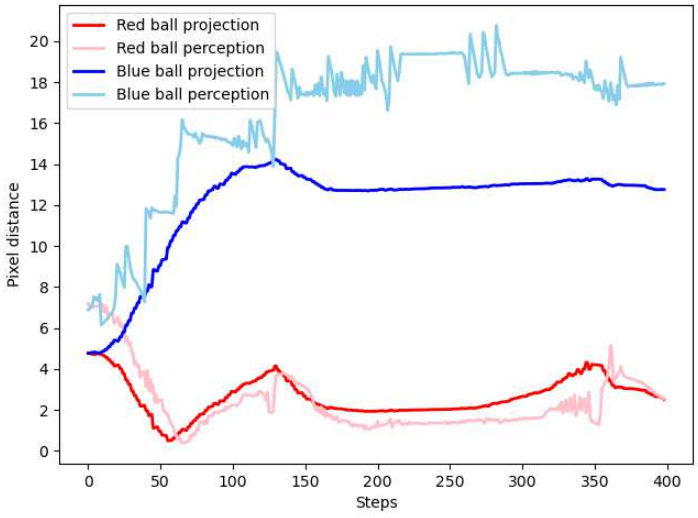

Figure 5.13: Focusing on red ball with mode `mean`, and the blue ball remains in sight.

We can see that the `mean` mode maintains beliefs over both objects fairly well. However, when considering objects that are not constantly in sight, the model falls short, as shown in Fig. 5.14.

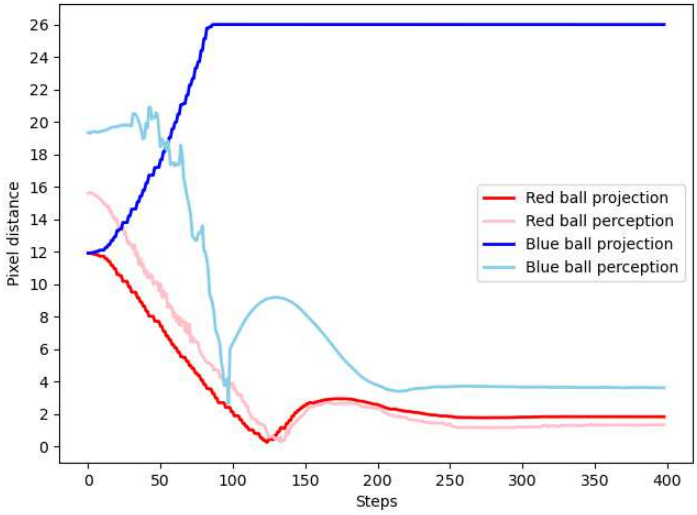

Figure 5.14: Focusing on red ball with mode `mean`, and the blue ball falls out of sight.

The belief over the red ball is maintained very well, without drift, but the blue ball belief quickly deteriorates after it is out of sight. The model explains its visual input by assuming the blue ball is somehow "hidden" behind the red ball, as seen in Fig. 5.15.

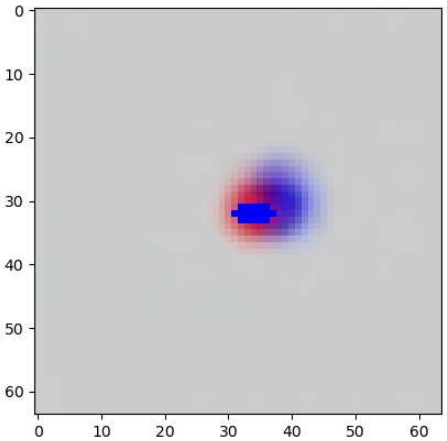

Figure 5.15: Visual prediction at end of trial. The absence of the blue ball is explained by it "merging" with or "hiding" behind the red ball.

The `mean` mode maintains belief over objects in sight without drift, but quickly forgets about the object out of sight. From both of these previous scenarios - using one of the modes exclusively - we can see that a better strategy is needed to accomplish a more stable

behavior. Table 5.1 shows the two modes compared by previously mentioned metrics.

Table 5.1: EXCLUSIVE MODE COMPARISON

	mean		**closest**	
	visible	**invisible**	**visible**	**invisible**
Reach error	2.52	1.84	24.67	10.63
Reach time	45	107	64	100
Reach stability	0.82	0.55	6.78	2.84
Perception error (red)	0.03	0.5	23.34	9.62
Perception error (blue)	5.17	22.37	4.73	0.09
Perception stability (red)	0.5	0.15	6.52	2.29
Perception stability (blue)	1.68	5.07	1.84	3.06
OPM	0.38	0.12	0.29	0.69

We can see that `mean` is generally more stable in both its perceptions and reach, and that it does not experience drift at the end of trial. However, when it comes to perception error, stability and object permanence of an invisible object, the `closest` mode performs better. To utilize the positive aspects of each mode, a balance must be made. Different ratios of these modes during inference have been Appendix B. The ratio that achieves the best compromise between stability, accuracy and object permanence is 20%, that is - mode `mean` is used every fifth step, while `closest` is used for the rest.

5.4 Inference precisions and variables

The tuning of precisions - α, β, π_{visual} and k - is vital for successful operation of the model. We have considered the effects of these precisions on the focus task where model must focus on the red ball, while the blue ball falls out of sight. The baseline settings for all trials are in Appendix A, and the trial results in Appendix B. The configuration that showed itself best during trials is the baseline of $\alpha = 0.5$, $\beta = 1 \cdot 10^{-1}$, $\pi = 6 \cdot 10^{-3}$ and $k = 6 \cdot 10^{-2}$.

6 Discussion

6.1 Model behaviour

The active inference attention model developed in this thesis accomplishes multiple functions of visual attention. In this section we summarize the results and analyze the behavior of the model under different tasks.

Direction and maintenance of focus

The model is capable of maintaining belief over objects it sees in the environment and can direct its gaze on either of them depending on its sensory input. When focusing on one object, it maintains the belief over the other, even showing behaviour similar to object permanence. If an object goes out of sight, the model maintains belief over its location in the environment for some time. Direction and maintenance of focus is possible even when the objects are moving, provided that the agent can catch up to them.

Maintenance of the model's belief is highly dependant on the precisions from the downwards sensory input and from the upwards intentions. The mode of intention generation dictates how well an object out of sight is perceived. The mode `closest` successfully maintains belief over the invisible object, but causes a deterioration of the belief of the visible object and a drifting in the agent's action. The mode `mean`, on the other hand, quickly forgets about the invisible objects true location, but maintains the belief over the visible object. Using a correct balance of these two modes enables the model to maintain a good belief over an invisible object, while having a stable focus.

Shifting attention

Another key function that the model accomplishes is the ability to shift its focus from one object to the other, even when the other is out of sight. When considering static objects, the model successfully shifts its attention in the case where both objects are visible. In the case of one object going out of sight upon focusing on the first one, the model is capable of shifting focus - but its belief over the second object deteriorates after multiple shifts.

The model is more successful when holding focus for longer before an attention shift, which was the case in the trial with moving objects. The perceived locations of both objects were very close to the true ones, enabling better attention shifts. Once again, choosing a correct intention generation mode is crucial in maintaining beliefs over other objects in the environment.

6.2 Effects of precisions

This section discusses the effects of varying precisions. As before, we consider $\alpha, \beta, \pi_{visual}$ and k.

Sensory ratio α

The variable $\boldsymbol{\alpha}$ is responsible for regulating the sensory precision weights $\boldsymbol{\pi}$. It determines how much each sensory modality contributes to the belief update and action. During trials in which the objects are static, varying $\boldsymbol{\alpha}$ does not change the model's behaviour signifiantly. However, in trials with moving objects, an $\boldsymbol{\alpha}$ which favors the visual input more ($\boldsymbol{\alpha} < 0.5$) than the proprioceptive performs better. This is because of the more dynamic visual input, so the visual update needs to be "stronger".

Increasing $\boldsymbol{\alpha}$ to 1.0 completely shuts of the belief and action update from the proprioceptive input, which leads to drastic and unstable belief and action updates. This is because the visual gradients affecting action are quite noisy. This drastic fall in accuracy and stability can be seen in all trials with $\boldsymbol{\alpha}$ close or equal to 1.

Intention multiplier β and gain k

The intention multiplier does not play a major role in action accuracy, but serves as a stabilizer of belief and either suppresses belief update through intentions or increases the intention attractions. The gain k plays a similar function in the equations of system dynamics, because the errors k multiplies act as attractors to the belief update. From the trials w can conclude that these variables play a minor role in accuracy, but a bigger role in object permanence.

Visual sensory precision π_{visual}

The main purpose of $\boldsymbol{\pi_{visual}}$ is to lesser the magnitude of visual gradients from the decoder network. The trials show that a high $\boldsymbol{\pi_{visual}}$ produces unstable behaviours because of the gradient magnitude, while a lower $\boldsymbol{\pi_{visual}}$ leads to the model failing to adapt to the changing environment.

6.3 Shortcomings

One shortcoming of the developed model is that it is capable of encoding belief for only two objects in the environment. A more complex model would be able to handle multiple objects, even when they are of the same type. This would require a more complex encoding than used here, but would enable more complex attention mechanisms.

Another shortcoming is the shape of the visual belief, which is the latent space of the VAE. In our case this space was forced to have a certain structure that is easily interpreted, but it required semi-supervised training of the VAE. A positive of this interpretable structure is that the mapping from visual sensory input to action is trivial and does not have to be learned, whereas an unsupervised latent space would require learning this mapping. A more complex model with more sensory modalities would need to have labeled data for every modality, which is hard to gather when the agent is not in simulation.

7 Conclusion

The goal of this thesis was to develop and implement an active inference model of visual attention. The developed model uses the decoder part of a VAE as a generative model for the visual sensory input, while simple generative models were used for proprioceptive input and needs. The model is capable of holding belief over two objects in the environment and can generate intentions that guide its focus on those objects. A simple attention mechanism is responsible for the choice of object.

The model, implemented in Python and ROS, was capable of directing and maintaining focus on both static and moving objects in the environment. It was also able to shift its focus from one object onto the other, also in the cases of static and moving objects. One ability that emerged is not unlike the phenomenon of object permanence - the model maintains belief over objects that are out of sight for some time, which enables it to shift attention to an object it previously saw, but does not see now.

Future goals might be to implement more complex attention that would choose objects based on multiple features (color, shape, velocity, etc.), as well as integrating more sensory modalities that could influence attention between each other. Unsupervised and reinforcement learning would be used for easier model training.

References

[1] D. C. Knill and A. Pouget, "The bayesian brain: the role of uncertainty in neural coding and computation," *Trends in Neurosciences*, vol. 27, no. 12, pp. 712–719, 2004. https://doi.org/https://doi.org/10.1016/j.tins.2004.10.007

[2] K. Friston, "The free-energy principle: a unified brain theory?" *Nature Reviews Neuroscience*, vol. 11, no. 2, pp. 127–138, Feb 2010. https://doi.org/10.1038/nrn2787

[3] K.Friston, "The history of the future of the bayesian brain," *NeuroImage*, vol. 62, no. 2, p. 1230–1233, Aug. 2012. https://doi.org/10.1016/j.neuroimage.2011.10.004

[4] K. Friston, J. Kilner, and L. Harrison, "A free energy principle for the brain," *Journal of Physiology-Paris*, vol. 100, no. 1, pp. 70–87, 2006, theoretical and Computational Neuroscience: Understanding Brain Functions. https://doi.org/https://doi.org/10.1016/j.jphysparis.2006.10.001

[5] R. Born and G. Bencomo, "Illusions, Delusions, and Your Backwards Bayesian Brain: A Biased Visual Perspective," *Brain Behavior and Evolution*, vol. 95, no. 5, pp. 272–285, 07 2021. https://doi.org/10.1159/000514859

[6] D. Rahnev, "The bayesian brain: What is it and do humans have it?" *Behav. Brain Sci.*, vol. 42, no. e238, p. e238, Nov. 2019.

[7] R. P. N. Rao and D. H. Ballard, "Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects," *Nature Neuroscience*, vol. 2, no. 1, p. 79–87, Jan. 1999. https://doi.org/10.1038/4580

[8] Y. Nagai, "Predictive learning: its key role in early cognitive development," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 374, no.

1771, p. 20180030, Mar. 2019. https://doi.org/10.1098/rstb.2018.0030

[9] K. Friston and S. Kiebel, "Predictive coding under the free-energy principle," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 364, no. 1521, p. 1211–1221, May 2009. https://doi.org/10.1098/rstb.2008.0300

[10] M. Corbetta and G. L. Shulman, "Control of goal-directed and stimulus-driven attention in the brain," *Nature Reviews Neuroscience*, vol. 3, no. 3, p. 201–215, Mar. 2002. https://doi.org/10.1038/nrn755

[11] K. J. Friston, J. Daunizeau, and S. J. Kiebel, "Reinforcement learning or active inference?" *PLoS ONE*, vol. 4, no. 7, p. e6421, Jul. 2009. https://doi.org/10.1371/journal.pone.0006421

[12] K. Friston, J. Mattout, N. Trujillo-Barreto, J. Ashburner, and W. Penny, "Variational free energy and the laplace approximation," *NeuroImage*, vol. 34, no. 1, p. 220–234, Jan. 2007. https://doi.org/10.1016/j.neuroimage.2006.08.035

[13] T. Parr and K. J. Friston, "Active inference and the anatomy of oculomotion," *Neuropsychologia*, vol. 111, p. 334–343, Mar. 2018. https://doi.org/10.1016/j.neuropsychologia.2018.01.041

[14] M. Priorelli, G. Pezzulo, and I. P. Stoianov, "Active vision in binocular depth estimation: A top-down perspective," *Biomimetics*, vol. 8, no. 5, p. 445, Sep. 2023. https://doi.org/10.3390/biomimetics8050445

[15] T. Rood, M. van Gerven, and P. Lanillos, *A Deep Active Inference Model of the Rubber-Hand Illusion*. Springer International Publishing, 2020, p. 84–91. https://doi.org/10.1007/978-3-030-64919-7_10

[16] M. Priorelli and I. P. Stoianov, *Efficient Motor Learning Through Action-Perception Cycles in Deep Kinematic Inference*. Springer Nature Switzerland, Nov. 2023, p. 59–70. https://doi.org/10.1007/978-3-031-47958-8_5

[17] C. Sancaktar, M. A. J. van Gerven, and P. Lanillos, "End-to-end pixel-based deep active inference for body perception and action," in *2020 Joint IEEE 10th International Conference on Development and Learning and Epigenetic Robotics*

(ICDL-EpiRob). IEEE, Oct. 2020. https://doi.org/10.1109/icdl-epirob48136.2020.9278105

[18] G. Oliver, P. Lanillos, and G. Cheng, "An empirical study of active inference on a humanoid robot," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 14, no. 2, p. 462–471, Jun. 2022. https://doi.org/10.1109/tcds.2021.3049907

[19] R. Desimone, "Neural mechanisms for visual memory and their role inattention," *Proceedings of the National Academy of Sciences*, vol. 93, no. 24, p. 13494–13499, Nov. 1996. https://doi.org/10.1073/pnas.93.24.13494

[20] L. Itti and P. Baldi, "Bayesian surprise attracts human attention," *Vision Research*, vol. 49, no. 10, p. 1295–1306, Jun. 2009. https://doi.org/10.1016/j.visres.2008.09.007

[21] E. Holmes, T. Parr, T. D. Griffiths, and K. J. Friston, "Active inference, selective attention, and the cocktail party problem," *Neuroscience & Biobehavioral Reviews*, vol. 131, p. 1288–1304, Dec. 2021. https://doi.org/10.1016/j.neubiorev.2021.09.038

[22] M. B. Mirza, R. A. Adams, K. Friston, and T. Parr, "Introducing a bayesian model of selective attention based on active inference," *Scientific Reports*, vol. 9, no. 1, Sep. 2019. https://doi.org/10.1038/s41598-019-50138-8

[23] T. Rohe and U. Noppeney, "Cortical hierarchies perform bayesian causal inference in multisensory perception," *PLOS Biology*, vol. 13, no. 2, p. e1002073, Feb. 2015. https://doi.org/10.1371/journal.pbio.1002073

[24] A. J. Yu and P. Dayan, "Uncertainty, neuromodulation, and attention," *Neuron*, vol. 46, no. 4, p. 681–692, May 2005. https://doi.org/10.1016/j.neuron.2005.04.026

[25] A. Ferrari and U. Noppeney, "Attention controls multisensory perception via two distinct mechanisms at different levels of the cortical hierarchy," *PLOS Biology*, vol. 19, no. 11, p. e3001465, Nov. 2021. https://doi.org/10.1371/journal.pbio.3001465

[26] M. Priorelli and I. P. Stoianov, "Flexible intentions: An active inference theory," *Frontiers in Computational Neuroscience*, vol. 17, Mar. 2023. https:

//doi.org/10.3389/fncom.2023.1128694

[27] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, “beta-VAE: Learning basic visual concepts with a constrained variational framework,” in *International Conference on Learning Representations*, 2017. [Online]. Available: https://openreview.net/forum?id=Sy2fzU9gl

[28] E. Mathieu, T. Rainforth, N. Siddharth, and Y. W. Teh, “Disentangling disentanglement in variational autoencoders,” 2018.

[29] pi tau, “vae,” https://github.com/pi-tau/vae, 2024, f5a3330.

Abstract

Goal-directed visual attention using a deep active inference model

Tin Mišić

Visual attention is a complex cognitive process that depends on both the internal state of an agent and the external environment. The agent must actively select parts of the sensory space to focus on to satisfy its internal needs. In this thesis we developed a deep learning model that uses visual and proprioceptive sensory input, directing attention to objects in the environment based on internal needs, thereby achieving complex goals. The model's development relies on the free-energy principle, utilizing predictive coding to predict sensory input and active inference for motor control and action execution. A generative model was implemented using a variational auto-encoder to predict visual sensory data. Extensive training and testing were conducted in a simulated environment to evaluate the model's performance. The results demonstrated that the model consistently directed attention to objects that satisfied its current needs and goals and could dynamically shift focus as the needs changed. With specific precision tunings, a behaviour similar to object permanence emerged. Furthermore, the model's architecture allows for the integration of additional sensory modalities, such as auditory or tactile inputs, which can enhance its ability to prioritize attention based on a broader range of sensory information.

Keywords: active inference; visual attention; variational auto-encoder

Sažetak

Ciljno-usmjerena vizualna pažnja koristeći duboki model aktivnog zaključivanja

Tin Mišić

Vizualna pažnja je složen kognitivni proces koji ovisi kako o unutarnjem stanju agenta, tako i o stanju okoline. Agent mora aktivno odabrati dijelove senzornog prostora na koje će se usmjeriti kako bi zadovoljio svoje unutarnje potrebe. U ovom radu razvili smo model dubokog učenja koji koristi vizualne i proprioceptivne senzorne ulaze, usmjeravajući pažnju na objekte u okruženju na temelju unutarnjih potreba, čime postiže složene ciljeve. Razvoj modela oslanja se na princip slobodne energije, koristeći prediktivno kodiranje za predviđanje senzornog unosa i aktivno zaključivanje za motoričku kontrolu i izvođenje akcija. Generativni model je implementiran korištenjem varijacijskog auto-enkodera za predviđanje vizualnih senzorskih podataka. Opsežno treniranje i testiranje provedeni su u simuliranom okruženju kako bi se procijenila učinkovitost modela. Rezultati su pokazali da je model dosljedno usmjeravao pažnju na objekte koji su zadovoljavali njegove trenutne potrebe i ciljeve te je mogao dinamički mijenjati fokus kako su se potrebe mijenjale. S određenim podešavanjima preciznosti, pojavilo se ponašanje slično trajnosti objekta. Nadalje, arhitektura modela omogućuje integraciju dodatnih senzornih modaliteta, kao što su slušni ili taktilni ulazi, što može poboljšati njegovu sposobnost prioritiziranja pažnje na temelju šireg spektra senzornih informacija.

Ključne riječi: aktivno zaključivanje; vizualna pažnja; varijacijski auto-enkoder

Appendix A: Trial setups

Focus on static objects

α	β	π	k	Int. Mode	Length	Attn. Period
0.5	$1 \cdot 10^{-1}$	$6 \cdot 10^{-3}$	$6 \cdot 10^{-2}$	50%	300	N/A

Focus on moving objects

α	β	π	k	Int. Mode	Length	Attn. Period
0.1	$7 \cdot 10^{-2}$	$6 \cdot 10^{-3}$	$6 \cdot 10^{-2}$	closest	300	N/A

Note: Objects are moving at a speed of 1 unit per 5 seconds.

Attention shift on static objects

α	β	π	k	Int. Mode	Length	Attn. Period
0.5	$1 \cdot 10^{-1}$	$6 \cdot 10^{-3}$	$6 \cdot 10^{-2}$	closest	300	100

Attention shift on moving objects

α	β	π	k	Int. Mode	Length	Attn. Period
0.25	$7 \cdot 10^{-2}$	$6 \cdot 10^{-3}$	$6 \cdot 10^{-2}$	50%	600	300

Note: Objects are moving at a speed of 1 unit per 5 seconds.

`closest` and `mean` comparison

α	β	π	k	Int. Mode	Length	Attn. Period
0.5	$1 \cdot 10^{-1}$	$6 \cdot 10^{-3}$	$6 \cdot 10^{-2}$	**varies**	400	N/A

Precision values comparison - baseline

α	β	π	k	Int. Mode	Length	Attn. Period
0.5	$1 \cdot 10^{-1}$	$6 \cdot 10^{-3}$	$6 \cdot 10^{-2}$	50%	400	N/A

Appendix B: Mode and precision trial results

B.1 Mode ratios

We have explored using a ratio of these modes, where m represents the ratio of steps that use mode `mean` over steps that use mode `closest`. We examined the model's behavior for the cases $m \in \{0.1, 0.1429, 0.2, 0.25, 0.5\}$. The trial results can be seen in Fig. B.1.

B.2 Inference precisions and variables

All of the trials use the baseline values in Appendix A, with only the target variable changing.

B.2.1 Reach

The reach time for the different values of the variables can be seen in Fig. B.4.

We can see that the reach error is large for the higher values of all variables, but stays fairly constant for the lower values. The exception is π_{visual}, which also grows the lower its value. This can be explained by the model putting less weight on the visual input and therefore trusting its current belief more, which prevents it from reaching the target. The higher values make increase the gradients and attractive forces, making the model behave erratically, which prevents accurate reach.

Similar results are achieved for reach stability (Fig. B.3), while reach time stays fairly consistent and is mainly dependant on the speed of turning of the agent (Fig. B.4.

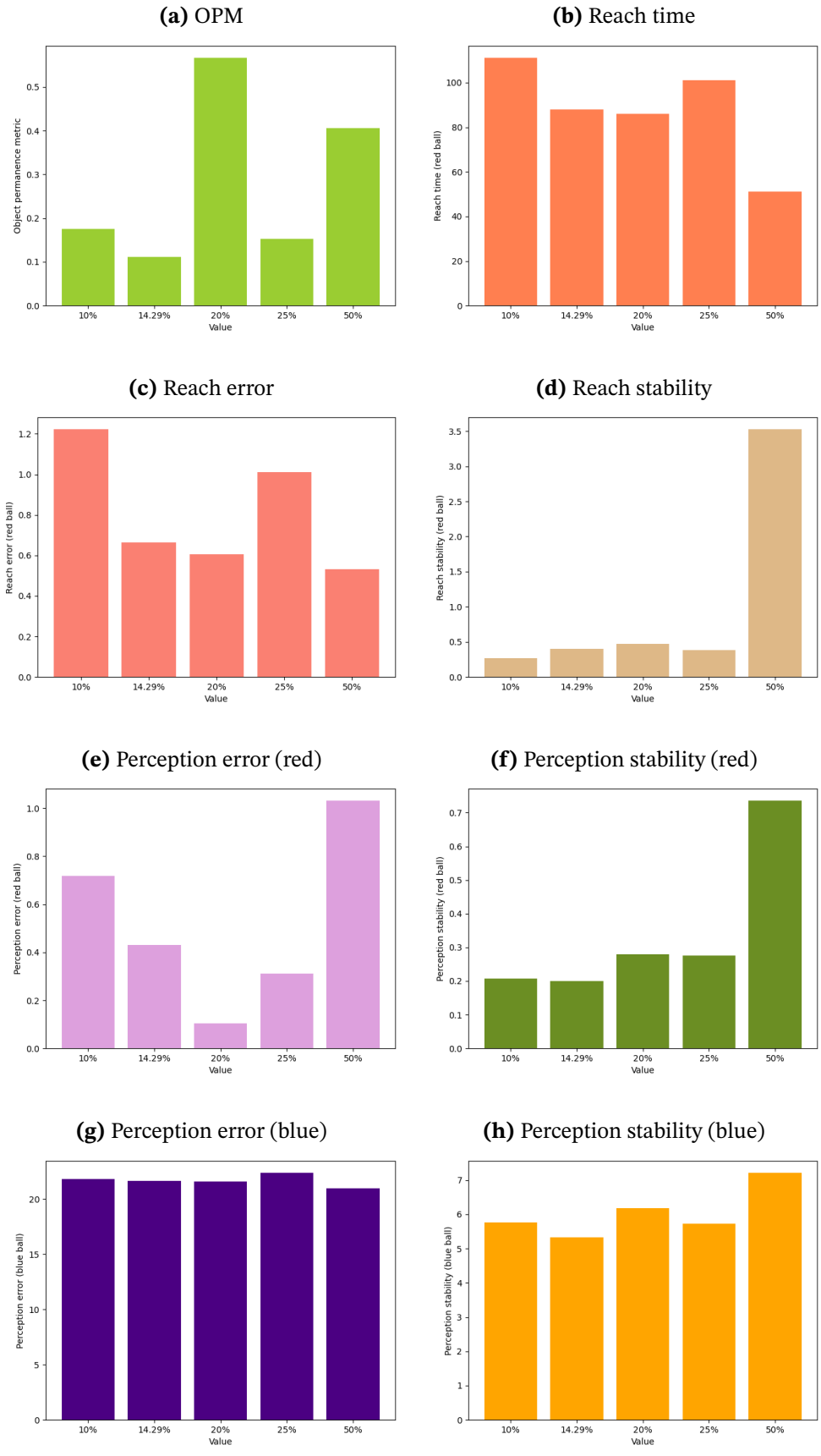

Figure B.1: Mode ratio trial results

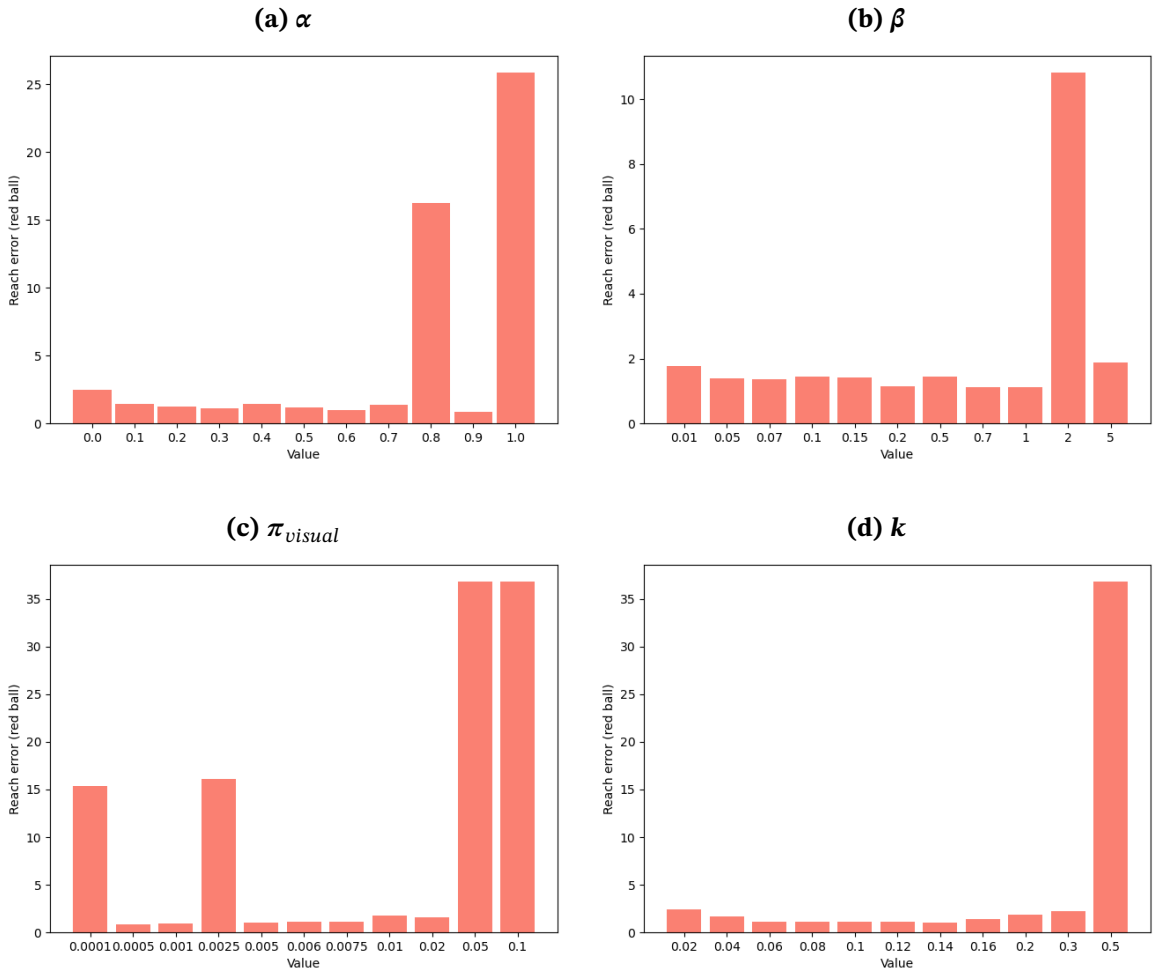

Figure B.2: Reach error for different values of α, β, π_{visual} and k

B.2.2 Perception

Perception errors for the red ball (the one being focused on) under varying precisions can be seen in Fig. B.5. Perception errors for the blue ball (the one that goes out of sight) are in Fig. B.6.

We can see that the perception error for the red ball is very similar to the reach error, while the perception of the blue ball remains fairly constant. The object permanence metric, however, varies from value to value, and can be seen in B.7.

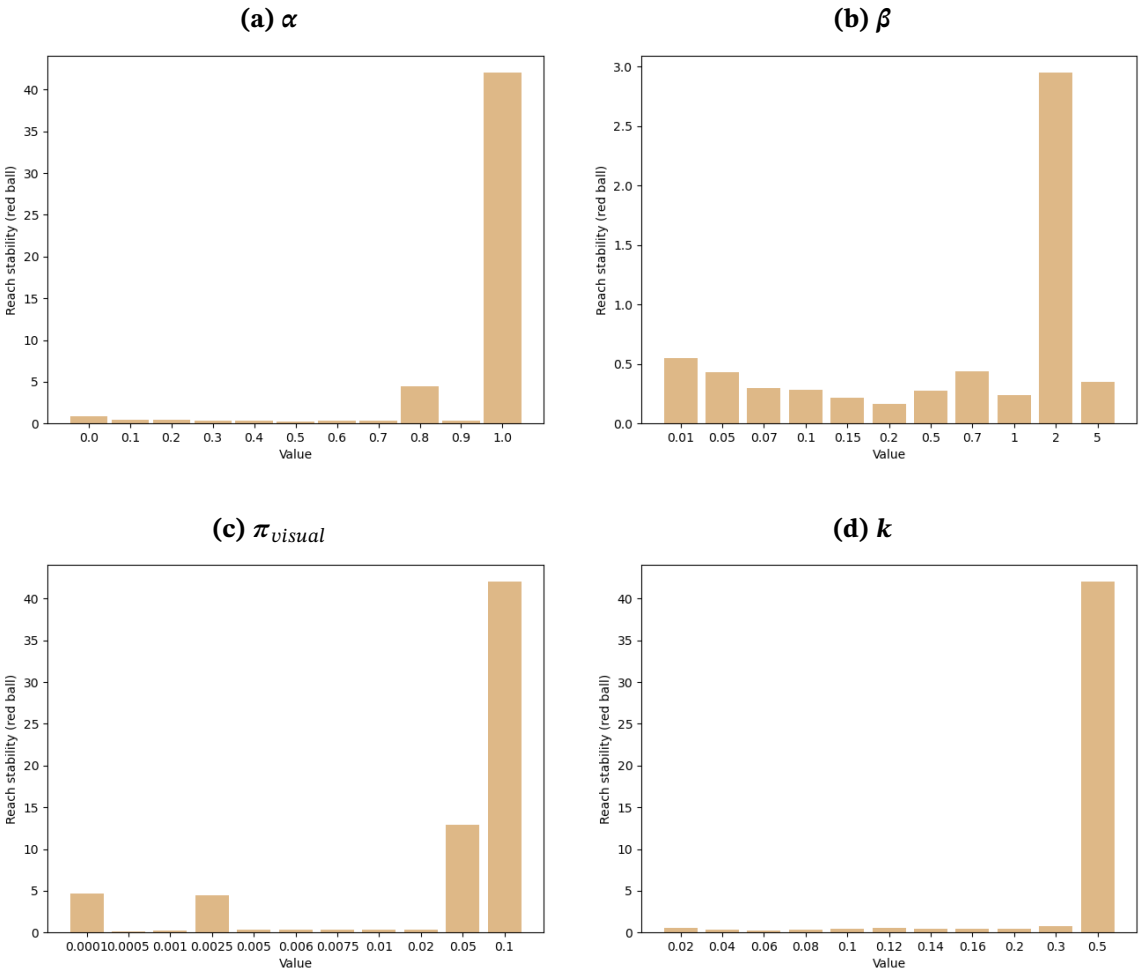

Figure B.3: Reach stability for different values of α, β, π_{visual} and k

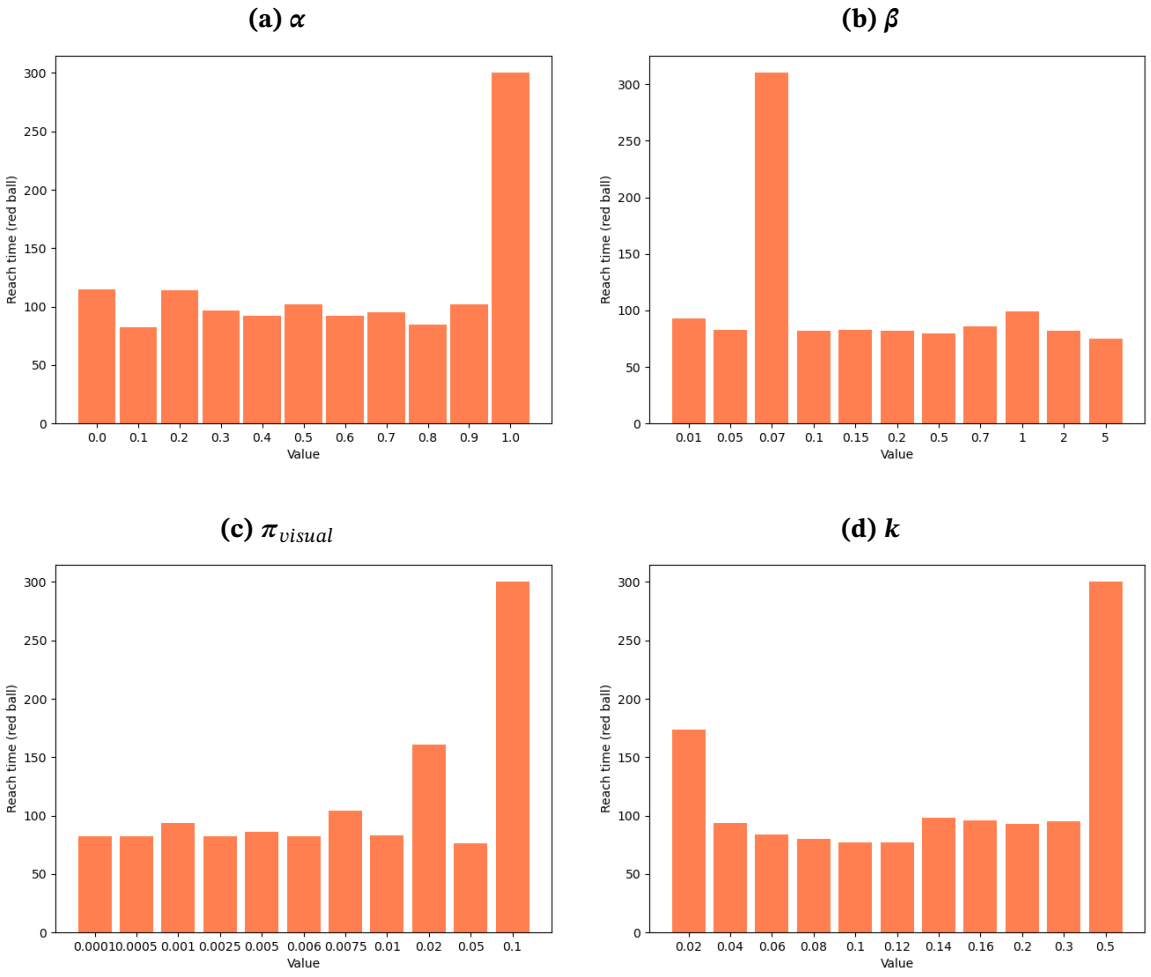

Figure B.4: Reach time for different values of α, β, π_{visual} and k

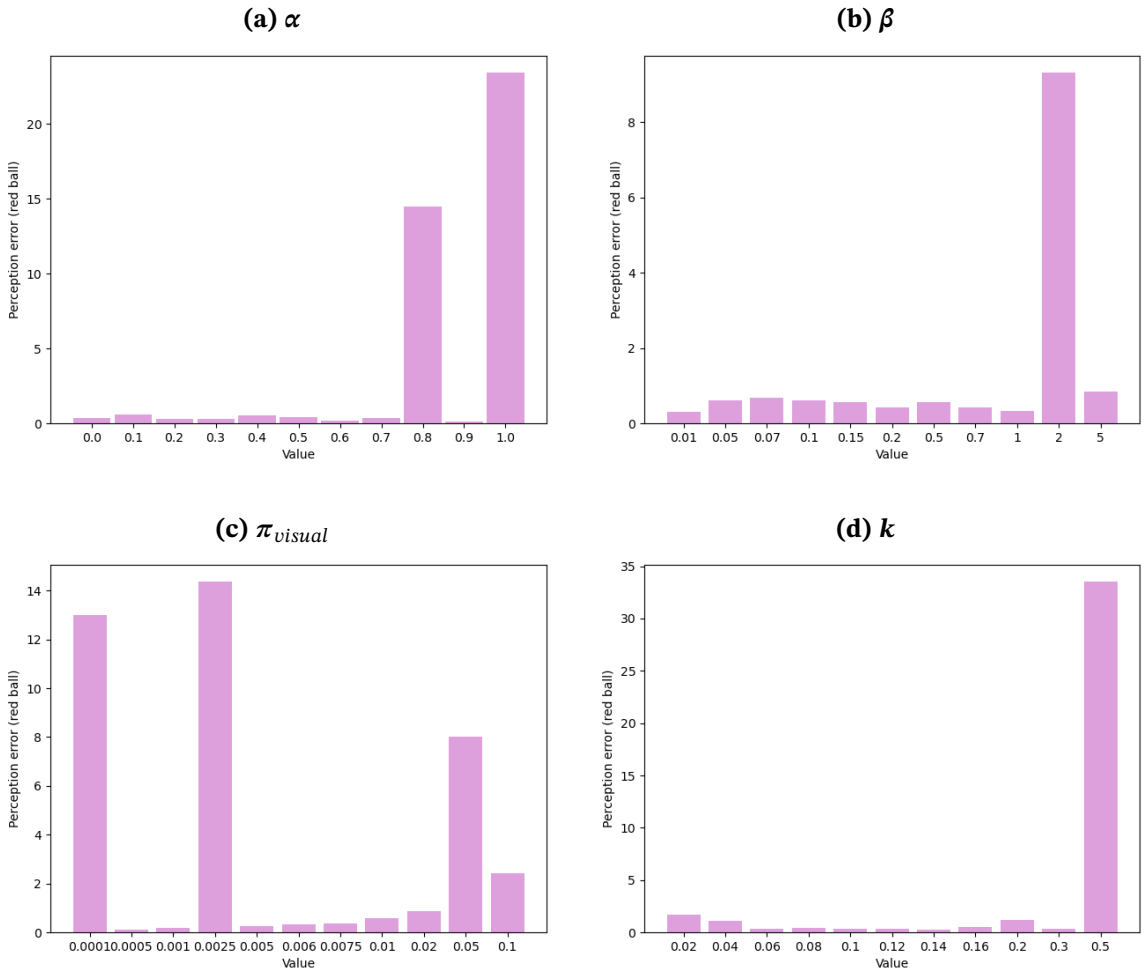

Figure B.5: Perception error of red ball for different values of α, β, π_{visual} and k

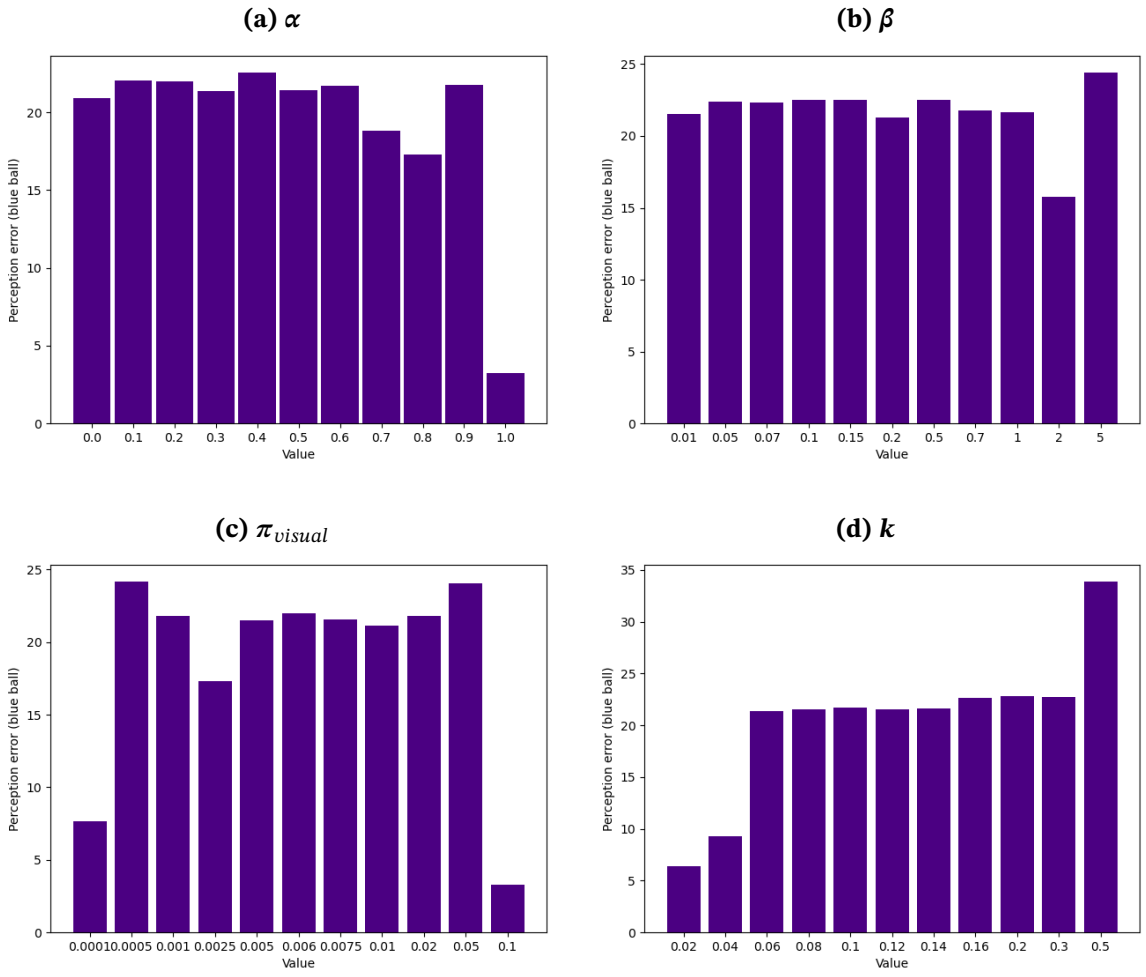

Figure B.6: Perception error of blue ball for different values of α, β, π_{visual} and k

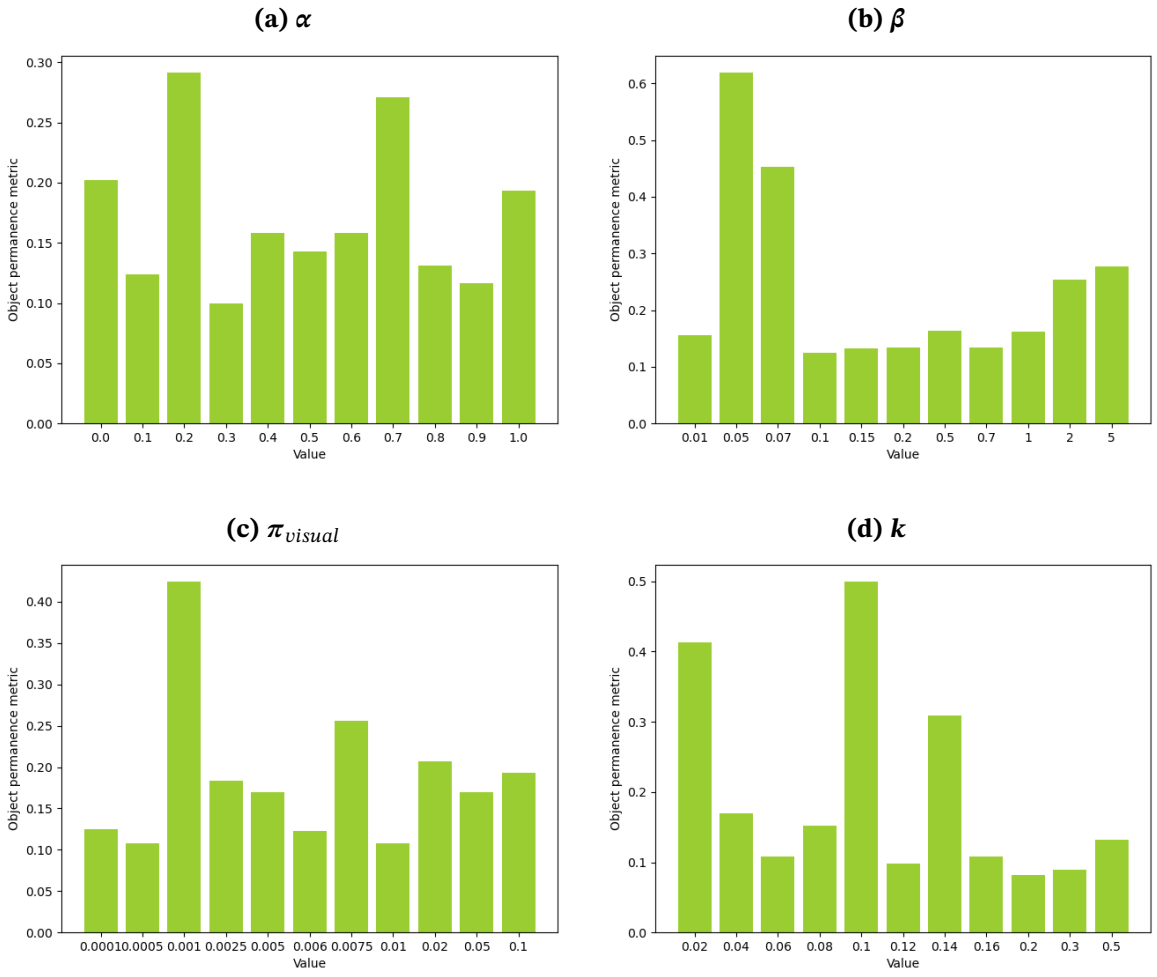

Figure B.7: Object permanence metric for different values of α, β, π_{visual} and k