2011

A cortical model of object perception based on Bayesian networks and belief propagation.

Dura-Bernal, Salvador

A cortical model of object perception based on Bayesian networks and belief propagation.

Salvador Durá Bernal

PhD Thesis in Computational Neuroscience

2010

A cortical model of object perception based on Bayesian networks and belief propagation.

Salvador Dura Bernal.

Centre for Robotics and Neural Systems.
School of Psychology.

University of Plymouth

A thesis submitted to the University of Plymouth in partial fulfillment of the requirements for the degree of:

Philosophy Doctor in Computational Neuroscience.

October 2010.

Abstract

A cortical model of object perception based on Bayesian networks and belief propagation.

Salvador Dura Bernal.

Evidence suggests that high-level feedback plays an important role in visual perception by shaping the response in lower cortical levels (Sillito et al. 2006, Angelucci and Bullier 2003, Bullier 2001, Harrison et al. 2007). A notable example of this is reflected by the retinotopic activation of V1 and V2 neurons in response to illusory contours, such as Kanizsa figures, which has been reported in numerous studies (Maertens et al. 2008, Seghier and Vuilleumier 2006, Halgren et al. 2003, Lee 2003, Lee and Nguyen 2001). The illusory contour activity emerges first in lateral occipital cortex (LOC), then in V2 and finally in V1, strongly suggesting that the response is driven by feedback connections. Generative models and Bayesian belief propagation have been suggested to provide a theoretical framework that can account for feedback connectivity, explain psychophysical and physiological results, and map well onto the hierarchical distributed cortical connectivity (Friston and Kiebel 2009, Dayan et al. 1995, Knill and Richards 1996, Geisler and Kersten 2002, Yuille and Kersten 2006, Deneve 2008a, George and Hawkins 2009, Lee and Mumford 2003, Rao 2006, Litvak and Ullman 2009, Steimer et al. 2009).

The present study explores the role of feedback in object perception, taking as a starting point the HMAX model, a biologically inspired hierarchical model of object recognition (Riesenhuber and Poggio 1999, Serre et al. 2007b), and extending it to include feedback connectivity. A Bayesian network that captures the structure and properties of the HMAX model is developed, replacing the classical deterministic view with a probabilistic interpretation. The proposed model approximates the selectivity and invariance operations of the HMAX model using the belief propagation algorithm. Hence, the model not only achieves successful feedforward recognition invariant to position and size, but is also able to reproduce modulatory effects of higher-level feedback, such as illusory contour completion, attention and mental imagery. Overall, the model provides a biophysiologically plausible interpretation, based on state-of-the-art probabilistic approaches and supported by current experimental evidence, of the interaction between top-down global feedback and bottom-up local evidence in the context of hierarchical object perception.

Contents

List of Figures

Acknowledgements

First of all, I would like to dedicate this thesis to the memory of my mum and dad for their unconditional love and for making me who I am. I would have never got here without them so this thesis is also theirs.

You meet many people along the four-year journey of a PhD. Here I would like to thank some of the ones that have played an important role in my life at some point during this period.

I would like to give my most sincere thanks to my supervisor Sue Denham for believing in me all they way through, for her excellent guidance, motivation, support and deep involvement in my research. Despite being extremely busy she always found time for me and I knew her door was always open. Thanks also to my second supervisor Thomas Wennekers for his advice and support, his closeness and all those late chats full of German sarcastic humour. I am also very grateful to Cathy for her thorough proof-reading of the entire thesis and excellent comments (sorry for any possible side-effects).

Many thanks to Andy, Floor, Lucy, Beibei, Jonathan, Emili and Martin (a life-saving Latex guru!) for helping me so much during my first scary year and for making the time at the red office much more enjoyable. Thanks also to James, Tim, Joao, Alex, Cathy, Giovanni and John for making the last year in the same old office feel like a warm and fun new place.

I need to thank all the friends that I've made in Plymouth for all their support and good times, especially Zuska, Jakub, Nidia, Dean and Martoula. Of course, I also need to thank my friends from Spain, who have always been there through the years and I consider as part of my family: Guille, Wito, Marrano, Katsu, Pino, Aguirre, Marta, Santaolalla, Cathy, Noe and many others who know who they are.

This thesis wouldn't have been possible without the love and support of my sister Cristina, my brother Pedro and my nephews and nieces Luis, Maria, Paula and Juan. They are always there to listen to me, give me advice and make me smile. Many thanks also to Eli for everything she has done.

Finally, I want to thank Ampi (La Pava) for having so much patience, listening to me every day (by now she must be an expert on Bayesian networks), making me laugh, making me happy and reminding me every day of the things that really matter in life. She made the thesis move to a second plane every day and thanks to that I have been able to finish it.

Authors declaration

At no time during the registration for the degree of Doctor of Philosophy has the author been registered for any other University award.

This study was financed with the aid of a scholarship from the University of Plymouth.

Relevant scientific seminars and conferences were regularly attended at which work was often presented. Two book chapters have been accepted for publication and one article has been submitted to a refereed journal.

Signed: 

Date: 27/08/11

Book chapters :

2010 *Dura-Bernal S*, Wennekers T, Denham SL
The role of feedback in a hierarchical model of object perception.
Brain Inspired Cognitive Systems 2010. Springer.

2010 Denham SL, *Dura-Bernal S*, Coath M, Balaguer-Ballester E
Neurocomputational Models of Perceptual Organization
Unconscious memory representation in perception: Processes and mechanisms in the brain, edited by Winkler I, Czigler I.

Conference papers (posters and talks) :

2010 *Dura-Bernal S. The role of feedback in a hierarchical model of object perception.*. **Probabilistic Neural Computation Group, Sheffield University** (Talk).

2010 *Dura-Bernal S*, Wennekers T, Denham SL. *Large-scale models of the visual cortical systems.*. **COLAMN, From Cortical Microcircuits to Brain-Inspired Computing, Plymouth** (Talk).

2010 *Dura-Bernal S*, Wennekers T, Denham SL. *The role of feedback in a hierarchical model of object perception.*. **Proceedings of the Brain Inspired Cognitive Systems conference, BICS'2010, Madrid** (Talk).

2009 *Dura S*, Wennekers T, Denham SL. *Feedback in a hierarchical model of object recognition in cortex.* **Proceedings of the Eighteenth Annual Computational Neuroscience Meeting, CNS'2009, Berlin.** BMC Neuroscience 10:P355 (Poster).

2009 *Durabernal S*, Wennekers T, Denham SL. *The role of cortical feedback in a hierarchical model of object perception.* **Proceedings of the 32nd European Conference on Visual Perception, ECVP'2009, Regensburg.** Perception 38 ECVP Abstract Supplement:134 (Poster).

2009 *Dura-Bernal S. The role of cortical feedback in a model of object perception.* **School of Psychology conference, University of Plymouth, 2009** (Talk).

2008 *Dura S*, Denham SL. *Feedback in a Hierarchical Model of Object Recognition: A Bayesian Inference Approach.*. **Local-Area Systems and Theoretical Neuroscience Day, LSTN'08, UCL, London** (Poster).

Word count for the main body of this thesis: $\sim 55,000$

Chapter 1

Introduction

1.1 Overview

Visual perception is a complex and largely unexplained process, which involves making sense of two-dimensional ambiguous retinal images by taking into account contextual and prior knowledge about the world (Friston 2005, Hochstein and Ahissar 2002, Gilbert and Sigman 2007). Although, traditionally, models of the visual system have focused on feedforward processes, it is becoming increasingly clear these are limited in capturing the wide range of complexities involved in visual perception. Recent reviews (Carandini et al. 2005, Olshausen and Field 2005) suggest that approximately only 20% of the response of a V1 neuron is determined by conventional feedforward pathways, while the rest arises from horizontal and feedback connectivity.

Anatomically, feedforward sensory pathways are paralleled by a greater number of top-down connections, which provide lower areas with massive feedback from higher cortical areas (Felleman and Van Essen 1991). Feedback terminations in the primary visual cortex (V1) are functionally organized and well-suited to centre-surround interactions, and unlike horizontal connections, their spatial and temporal properties have been found to provide an explanation for extra-classical distal surround effects (Angelucci and Bullier 2003). Experimental evidence shows that feedback originating in higher-level areas, such as V4, inferotemporal (IT) cortex or middle temporal (MT) cortex with bigger and more complex receptive fields, can modify and shape V1 responses, accounting for contextual or extra-classical receptive field effects (Hupe et al. 2001, Lee and Nguyen 2001, Murray et al. 2004, Sillito et al. 2006, Sterzer et al. 2006, Huang et al. 2007).

A notable example is observed in V1/V2 activity in response to illusory contours with no direct

retinal stimulation (e.g. Kanizsa figures), as reported in functional magentic resonance imaging (fMRI) (Maertens et al. 2008), electroencephalography (EEG) (Seghier and Vuilleumier 2006), magnetoencephalography (MEG) (Halgren et al. 2003) and single-cell recording (Lee 2003, Lee and Nguyen 2001) studies. The experiments show illusory contour-related activity emerging first in Lateral Occipital Cortex (LOC), then in V2 and finally in V1, strongly suggesting that the response is driven by feedback (Lee and Nguyen 2001, Murray et al. 2002).

While there is relative agreement that feedback connections play a role in integrating global and local information from different cortical regions to generate an integrated percept (Bullier 2001, Lee 2003), several differing approaches have attempted to explain the underlying mechanisms. Generative models and the Bayesian brain hypothesis provide a framework that can quantitatively model the interaction between prior knowledge and sensory evidence, in order to represent the physical and statistical properties of the environment. This framework provides an elegant interpretation of how bottom-up and top-down information across different cortical regions can be combined to obtain an integrated percept.

Increasing evidence supports the proposal that Bayesian inference provides a theoretical framework that maps well onto cortical connectivity, explains both psychophysical and neurophysiological results, and can be used to build biologically plausible models of brain function (Friston and Kiebel 2009, Dayan et al. 1995, Knill and Richards 1996, Geisler and Kersten 2002, Kording and Wolpert 2004, Yuille and Kersten 2006, Deneve 2008a). Within this framework, Bayesian networks and belief propagation provide a rigorous mathematical implementation of these principles. Belief propagation has been found to be particularly well-suited for neural implementation, due to its hierarchical distributed organization and homogeneous internal structure and operations (George and Hawkins 2009, Lee and Mumford 2003, Rao 2006, Litvak and Ullman 2009, Steimer et al. 2009).

The present study explores the role of feedback in object perception, taking as a starting point the HMAX model, a biologically inspired hierarchical model of object recognition (Riesenhuber and Poggio 1999, Serre et al. 2007b), and extending it to include feedback connectivity. By replacing the classical deterministic view with a probabilistic interpretation, a Bayesian net-

work that captures the structure and properties of the HMAX model is described. The proposed model also approximates the selectivity and invariance operations of the HMAX model using the belief propagation algorithm. Hence, the model not only achieves successful feedforward recognition invariant to position and size, but is also able to reproduce modulatory effects of higher-level feedback on lower-level activity, such as illusory contour completion.

The organization of this thesis is as follows. Chapter 2 reviews current evidence, theories and computational models of object perception. A special emphasis is placed on those that suggest moving from serial feedforward models towards more global and integrated approaches with feedback-mediated interactions between cortical regions.

Chapter 3 introduces generative models and the Bayesian brain hypothesis, providing a significant body of evidence that substantiates this approach. In the same chapter, Bayesian networks and belief propagation are described in detail, including an illustrative example. Existing computational models and plausible biological and cortical implementations are also reviewed.

Chapter 4 describes the methodology followed to develop the proposed model. This includes the probabilistic interpretation of HMAX as a Bayesian network, the model architecture, learning methods and feedforward and feedback functionality. Additionally, the chapter also describes several approximations and sampling methods to deal with the large scale of the network, the combination of information from multiple parents and the loops present in the network.

Chapter 5 presents the simulation results for feedforward invariant categorization and feedback modulation, with a focus on illusory contour completion.

Chapter 6 provides a deeper analysis and discussion of the simulation results and of the use of Bayesian networks to model object perception. Additionally, this chapter describes the model in relation to experimental data and to previous computational models, and suggest future lines of research.

1.2 Main contributions

The main contributions of this study are as follows:

- A review and analysis of the literature regarding object perception, feedback connectivity, illusory contour completion, generative models, Bayesian networks and belief propagation. This includes a detailed comprehensive explanation of belief propagation in Bayesian networks with several novel and illustrative examples.

- A Bayesian network implementing loopy belief propagation that captures the structure and functionality of HMAX, a feedforward object recognition model, and extends it to include dynamic recurrent feedback.

- Specific approximations and sampling methods that allow for the integration of information in large-scale Bayesian networks with loops and nodes with multiple parents.

- Demonstration that the model can account for invariant object categorization, mimicking the ventral path functionality.

- Demonstration that the model can account qualitatively for illusory contour formation and other higher-level feedback effects such as priming, attention and mental imagery.

Chapter 2

Object perception in the visual cortex

This chapter is intended to provide the necessary background knowledge and context to understand the motivation and methodological approach employed in the thesis, as well as the relevance of the results and conclusions obtained. The work in this thesis extends an existing feedforward model of object recognition to include feedback. Thus, Section 2.1 describes the principles of object perception in the visual cortex, together with supporting experimental evidence and existing computational models. Although object perception has been typically characterized as a feedforward process, the crucial role of feedback connections in this process is now widely accepted and strongly supported by experimental findings. Section 2.2 reviews experimental evidence and theoretical interpretations of the role of feedback in the visual system. One of the most notable perceptual effects that has been attributed to feedback is the formation of subjective contours, i.e. illusory and occluded contours. The model proposed in this thesis offers a plausible explanation for this phenomenon and provides simulation results in support. The basis of subjective contour formation is therefore thoroughly explored in Section 2.3.

2.1 Object recognition

2.1.1 Principles and experimental evidence

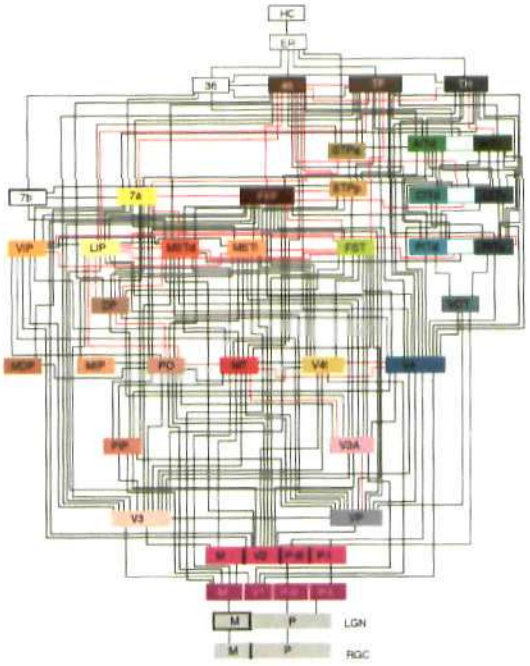
Object perception is an essential part of this thesis as it supplies the context and framework which is used to investigate and try to find answers to the research questions. These questions concern the functional role of feedback connections in the visual system, and, more precisely, along the object perception pathways. However, it is impractical to provide a comprehensive analysis of visual perception, as this is a vast area of research in itself, so this chapter is limited to covering the relevant aspects for this thesis. Therefore in this section, we start by describing

the principles underlying object recognition in the visual cortex, which can be understood as the initial feedforward processing stage leading to fast object categorization and identification (Serre et al. 2007b).

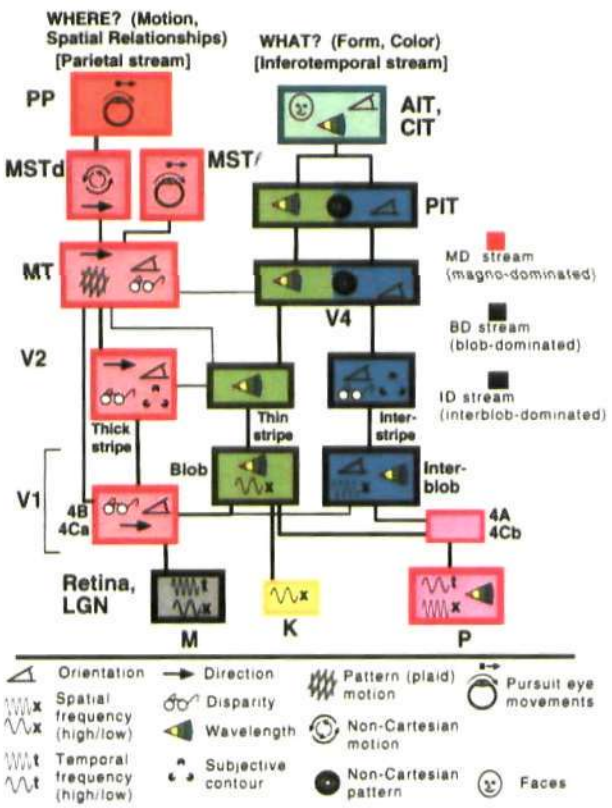
Note this section includes a general description of all the different areas involved in visual processing and an overview of the interactions which lead to an integrated percept. However, the section focuses on the feedforward processing strategies which lead to rapid object categorization, while Section 2.2 below deals specifically with feedback and the integration of information across the visual system.

2.1.1.1 Hierarchical distributed organization.

The visual system is capable of transforming light falling on the retina into neuronal electrical signals, which give rise to subjective visual perception. This is usually achieved in time periods measured in milliseconds, but requires complex information processing and encompasses several stages of analysis across many different regions. The macaque visual system, for example, has been classified into 32 distinct areas interconnected via over 300 reciprocal anatomical pathways (Felleman and Van Essen 1991), shown in Figure 2.1a. These areas have a hierarchical organization starting at the lateral geniculate nucleus (LGN), ascending through primary visual cortex (V1) and finishing in higher cortical structures. Each of these areas is considered to be functionally specialized, and embodies in itself a set of subdivisions (e.g. 6-layer cortical structure). Two major parallel processing streams have been identified in the visual hierarchy: the dorsal or *where* stream, and the ventral or *what* stream, schematically represented in Figure 2.1b (Van Essen and Gallant 1994).



(a)



(b)

Figure 2.1: For caption see footnote[1].

The dorsal stream originates in the magnocellular layers of the retina and LGN, then projects onto magno-dominated regions of V1 and upwards into the thick regions of secondary visual cortex (V2). A high incidence of cells selective for direction of motion have been found in these regions. The subsequent extrastriate areas involved are medial temporal (MT) cortex and medial superior temporal (MST) cortex, considered to be responsible for several motion analysis processes. These in turn project onto the posterior parietal (PP) cortex involved in higher level functions such as analyzing spatial relations and controlling eye movements. Overall the dorsal, or *where* pathway, which spreads along the parietal cortex, is mainly concerned with space, movement and action (Van Essen and Gallant 1994).

On the other hand, the ventral stream, or *what* pathway, is mostly concerned with object identification and perception. It encompasses the blob-dominated (BD) and the interblob-dominated (ID) parallel streams, both of which receive input mainly from parvocellular and koniocellular neurons in subcortical regions. The BD stream, which mediates color perception, originates in the blob compartments of V1 and feeds to the thin stripe regions of V2. The ID stream, responsible for form perception, projects from inter-blob compartments of V1 onto the inter-stripe regions of V2. Both streams converge onto the extrastriate regions V4 and IT cortex, which are associated with high-level functions of pattern recognition (Van Essen and Gallant 1994).

The complex interactions which exist between these two parallel processing pathways are still not well understood (Van Essen and Gallant 1994, Nassi and Callaway 2009). To make things

[1]Caption for Figure 2.1. a) Hierarchical structure of areas involved in visual processing in the macaque. The diagram shows 32 different areas linked by 187 connections, the majority of which are reciprocal pathways. It highlights the complexity and intricate interdependency of regions in the visual system (Felleman and Van Essen 1991). b) Functional schematic representation of the two hierarchical parallel visual processing streams (ventral/*what* and dorsal/*where*) in the macaque. Boxes represent visual regions, while lines represent the main connection pathways (usually bidirectional). Icons represent typical physiological and functional properties attributed to each region. The *where* path originates in the magnocellular layers of the retina and LGN (gray), then projects onto magno-dominated regions of V1 and upwards into the thick regions of V2 (red). Cells in these regions typically show selectivity to direction of motion. The subsequent extraestriate areas involved are MT and PP (red), responsible for several motion analysis processes. These in turn project onto the PP cortex (orange) involved in higher level functions such as analyzing spatial relations and controlling eye movements. The *what* path encompasses the blob-dominated (BD) (green) and the interblob-dominated (ID) (blue) parallel streams, both of which receive input mainly from parvocellular (P) (pink) and koniocellular (K) (yellow) neurons in subcortical regions. The BD stream, which mediates color perception, originates in the blob compartments of V1 and feeds to the thin stripe regions of V2. The ID stream, responsible for form perception, projects from inter-blob compartments of V1 onto inter-stripe regions of V2. Both streams converge onto the extraestriate regions V4 and IT cortex which are associated with high-level functions of pattern recognition (Van Essen and Gallant 1994). Pathways are partly based on anatomical information from Felleman and Van Essen (1991), represented in figure a). Note colors in a) do not correspond with colors in b).

even more complicated, there are also direct connections between regions separated by several hierarchical levels, such as V1 and MT (Felleman and Van Essen 1991). Further, each region may perform different visual processing functions at different times, requiring an intricate flow of information spanning multiple cortical areas (Bullier 2001, Lee et al. 1998). How this information is combined to generate a global percept, particularly for object perception, is discussed in Section 2.2, and is one of the key elements of this thesis. Therefore it is vital to provide a prior deeper analysis of the properties observed along the ventral processing stream.

2.1.1.2 Receptive field selectivity and invariance.

One of the main properties of the visual hierarchy concerns the selectivity of neurons at each level. The receptive field, or the stimulus which elicits the maximum response of a neuron, shows progressive increases in size and complexity as one ascends in the hierarchy (Figure 2.2).

Lateral geniculate nucleus (LGN) cells, which receive retinal input, respond to stimuli within relatively small concentric receptive fields with a center-surround organization. Within the visual system, LGN neurons are those whose response is better captured by existing models, even when using complex stimuli. These models have progressively been extended to include both linear and nonlinear components and gain-control mechanisms, as described later in this section. However, although these manage to predict a number of nonlinear phenomena (Carandini et al. 2005), they still fail to capture response properties emerging as a consequence of contextual modulation.

This is not surprising, firstly because even retinal cells, which project onto LGN and have been conventionally treated as simple prefilters for visual images, appear to be engaged in more complex computations, such as global motion detection (Gollisch and Meister 2010). Secondly, most LGN models focus on these feedforward retinal connections, which only account for approximately 15 % of the LGN cell input, whereas feedback connections, presumably involved in contextual processing, can account for over 30 % of their synaptic input.

V1 presents a much wider and more complicated distribution of receptive fields than retina and thalamus. Neurons in V1 can respond selectively to a variety of visual input attributes such as line orientation, direction of movement, contrast, velocity, colour, and spatial frequency. These

properties arise from retinal ganglion cells and LGN cells, which also exhibit some selectivity to contrast, velocity, colour and spatial frequency. V1 neurons with similar tuning properties tend to group together, leading to the classical columnar organization of ocular dominance and orientation preference in cortex (Hubel and Wiesel 1965).

Hubel and Wiesel (1965) were the first to propose the hierarchical organization of receptive fields, such that V1 simple cells are built from converging LGN cells aligned in space to produce the elongated *on-off* subregions observed. Additionally, the model provided the first classification of V1 cells, dividing them into *simple* and *complex*. Cells fell into the *simple* category if their receptive fields could be separated into *on* and *off* subregions, which could be linearly summated to predict the cell's response to different artificial stimuli. The rest of the cells, which did not have separate subregions, were categorized by exclusion as *complex* cells. However, the majority of V1 cells fall into the complex category. As will be described further down, there are also numerous variants within the simple and complex categories.

Several extensions improved the initial Hubel and Wiesel receptive field model of V1 neurons. Firstly, the linear filter was expanded to include a temporal dimension. Spatiotemporal receptive fields not only take into account the spatial profile, but also the temporal course of the response, and have proved to be crucial in understanding direction selectivity. This first filtering stage was shown to be well approximated by 2-dimensional Gabor filters (Jones and Palmer 1987). Secondly, a nonlinear stage was added, which described how the linear filter outputs were transformed into an instantaneous firing rate via a nonlinear Poisson process. The two-stage model was therefore called the linear-nonlinear (LN) model and provided a much better prediction of neuron responses than strictly linear filters, specially for retina and thalamic cells (Carandini et al. 2005).

Nonetheless, the model still had significant limitations (Ringach 2004). It was unable to account for the dependence on contrast of several response properties, such as saturation and summation size. For example, the greater the contrast of the stimulus, the smaller the degree of spatial summation, and thus the receptive field size. Furthermore, the LN model could not explain surround suppression, such as stimuli at an orthogonal orientation inhibiting the cells' response.

This led to an additional extension of the model to include gain control mechanisms, such that the output of the linear filter is divided by the overall activity of a *normalization pool*. The normalization pool typically includes cells in the near surround, but is not limited to those with similar tuning profiles, thus providing a *normalization* mechanism, which solved many of the previous limitations.

With respect to complex cells, the characterization of their receptive fields is less well understood and is still a topic of debate. Most models are derived from the original Hubel and Wiesel proposal and therefore assume complex receptive fields arise from combining the linear filters of a group of converging simple cells tuned to the same orientation. The most widespread example of this type of circuit is known as the energy model, which consists of two phase-shifted linear filters, tuned for orientation and spatial frequency, arranged in quadrature. The output of the filters is squared and then summed together to produce the response. Thus, the response will be high not only for images resembling the filters, but also for their inverses (Ringach 2004).

A recent study by Sasaki et al. (2010) analyzed the structure and and spatial relationship between the internal subunits and the overall receptive fields of complex cells. It concluded that complex cell subunits cannot be considered equivalent to simple cells, suggesting that complex cell receptive fields are constructed by a more a elaborate combination of linear filters than that proposed by Hubel and Wiesel. Alternative and more successful models of complex cell response, such as the *spike triggered covariance analysis*, provide a more accurate prediction of the cell's response to orientation and direction. This model is able to identify the different subunits present in the complex cell receptive field and quantify their contribution to the response of the cell (Carandini et al. 2005).

However, all existing models have been strongly influenced, and perhaps wrongly biased, by the original hierarchy model with two distinct neuron categories. As an alternative, it has been suggested that receptive fields in V1 lie along a continuum spectrum, with simple and complex cells at each end, allowing for additional cell types which would share properties of both simple and complex categories (Ringach 2004). A recent study, still in a preliminary stage, further challenges the classical model by suggesting cell response properties are a function of the type

of input employed (Fregnac 2010). Results showed the same neuron could exhibit simple or complex properties depending on whether the images presented dense or sparse noise.

It can easily be concluded that many crucial elements are still missing from current models of V1 response. Estimates suggest only 35% of the variance in natural images can be accounted for (Olshausen and Field 2005). A general point of agreement indicates the necessity to move beyond bottom-up filtering models to incorporate top-down feedback modulation as one of the basic components in any model of visual perception (Lee 2003, Olshausen and Field 2005, Carandini et al. 2005). This is not an easy task, as the response of neurons in higher visual processing areas is still very poorly understood.

The response properties of neurons in V2, which receive projections from area V1, are not nearly as well documented, and it is therefore uncertain what type of stimuli cause V2 neurons to respond optimally. Nonetheless, Hegde and Van Essen (2007) studied the responses of *a population of V2 neurons to complex contour and grating stimuli. They found several* V2 neurons responding maximally for features with angles, as well as for shapes such as intersections, tri-stars, fivepoint stars, circles, and arcs of varying length. Additionally, the receptive field sizes of V2 cells are approximately twice the size of those of V1. For example, at a retinal eccentricity of 2°, V1 receptive field size is $\sim 2^{\circ}$ of visual angle, while V2 receptive field size is $\sim 4^{\circ}$ (Angelucci et al. 2002). This is consistent with the hierarchical increase in the receptive field size and complexity proposed at the beginning of this section. Crucially, the increase of RF size implies a decrease in spatial resolution, which is a key aspect of the modelling study in this thesis.

Our current understanding of response selectivity in V4 neurons is also congruent with the hierarchical increase in size and complexity (Hegde and Van Essen 2007). However, at this level it is more difficult to characterize the exact receptive field of neurons, as these exhibit a wider range of preferred stimuli, and stronger invariance to stimulus transformations. Nevertheless, lesions of V4 in the macaque have caused impairments in pattern discrimination tasks (Van Essen and Gallant 1994). Further studies have shown V4 neurons can be tuned to shapes with specific type of boundary conformation at a given position within the stimulus, e.g. concave curvature

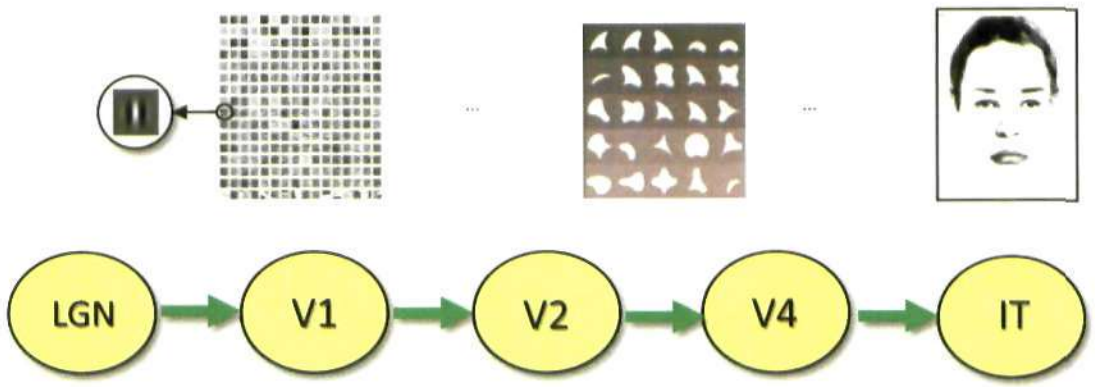
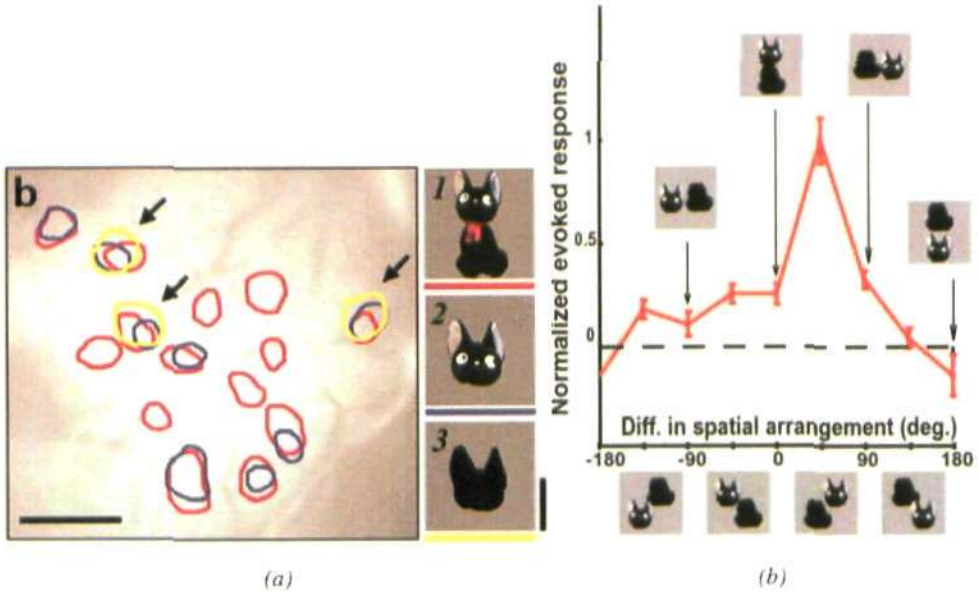


Figure 2.2: Idealized representation of the increase in receptive field size and complexity, from low-level to high-level areas of the visual system. Top-left: schematic representation of the small receptive fields tuned to simple features such as lines or edges, characteristic of V1. Top-middle: representation of typical V4 receptive fields, bigger and exhibiting more complex spatial profiles than in V1. Top-right: Schematic representation of large and invariant receptive fields in IT associated with objects, such as faces.

at the right, while being insensitive to other parts of the shape (Pasupathy and Connor 2001). Responses showed invariance to local transformations, such as small translations. Pasupathy and Connor (2002) also demonstrated how complete shapes were characterized as aggregates of boundary fragments represented by populations of V4 cells. This speaks for a representation of a complex stimulus in terms of its constituent parts. Therefore it can be argued that V4 response profiles roughly resemble shapes or small object parts of different complexities.

In the primate IT cortex neurons have been found to be selective to view-dependent representation of complex two-dimensional visual patterns, or objects such as faces or body parts (Logothetis et al. 1994, 1995). However, the way in which the objects are represented in IT is still an active area of research. Some results suggest objects are represented by a combination of cortical columns, each of which represents a visual feature, as depicted in Figure 2.3a. Others indicate not all columns are associated with a particular feature. A simpler object can sometimes be encoded by activating cortical columns that were not active for the more complex one. This suggests instead that objects arise as a combination of active and inactive columns, following *sparse coding* strategies (Tsunoda et al. 2001). Sparse coding refers to a type of neural code where each item is represented by the strong activation of a relatively small subset of



(a) (b)

Figure 2.3: a) Representation of complex objects in IT area, through the activation of cortical columns. In this example, simplified stimuli (cat's face silhouette) elicit only a subset of the regions evoked by the complex stimuli (complete cat) Tsunoda et al. (2001). b) Selectivity of cells in IT to different spatial arrangements of the parts of an object image. This suggests the spatial arrangement of object parts is also represented in IT (Yamane et al. 2006).

neurons. This could be related to the hierarchical categorical representation of IT populations, demonstrated recently by Kiani et al. (2007). In this study, cluster analysis was employed to show that IT populations' responses reconstruct part of our intuitive category structure, such as the global division into animate and inanimate objects, or faces which clustered into primate and non-primate faces.

Interestingly, Yamane et al. (2006) found neurons were sensitive to a particular spatial arrangement of the parts, which suggests an encoding of the spatial relationship between object parts (Figure 2.3b). The counterpart of the primate's IT in humans is thought to be the lateral occipital complex (LOC), which also exhibits a feature-based representation of the stimulus (Tanaka 1997). It has been associated with the representation of object parts (Hayworth and Biederman 2006), as well as with higher-level functions such as the perception of 3D volume from 2D shapes (Moore and Engel 2001).

Invariance to certain transformations of the input image also appears to be a prominent prop-

erty of higher-level neural responses. Population responses in IT have been shown to provide information about the identity and category of a novel object, generalizing over a range of positions and scales (Hung et al. 2005). Similar studies also show response invariance to the angle of view, and to intra-category identity (Hoffman and Logothetis 2009). Recently, Rust and DiCarlo (2010) demonstrated that both selectivity and invariance increased from V4 to IT. In general, higher levels in the perceptual hierarchy achieve higher degrees of invariance, such as view-invariant recognition of objects by interpolating between a number of stored views (Logothetis et al. 1994). Strikingly, Quiroga et al. (2005) demonstrated how neurons in hippocampus were able to respond selectively to more abstract concepts, such as 'the actress Halle Berry'. A particular neuron responded to drawings of her, herself dressed as Catwoman (a role she played in a movie), and to her written name. However, invariance to these attributes demonstrates an invariance beyond visual features.

2.1.2 Theoretical and computational models

Many computational models of object recognition exist in the literature. These can be divided into two broad categories: object-based and view-based. In the first group of models, the recognition process consists of extracting a view-invariant description of the object's structure which can then be compared to previously stored object descriptions. This can be done, for example, by decomposing the object into basic geometrical shapes which allows the structure of the object to be extracted independently of the viewpoint. The second category of models assumes objects are represented as a collection of view-specific features. The different views correspond to different image-based appearances due, for example, to different viewpoints or illuminations. These models usually rely on higher visual areas interpolating between several view-tuned units to create a view-invariant or object tuned response.

In this section, only those models relevant to this thesis are outlined. In particular the focus is placed on hierarchical view-based feedforward models constrained by the anatomical and physiological properties of the ventral path. These models usually span several regions of the visual cortex and therefore their biological realism is usually restricted to the network level of description.

2.1.2.1 HMAX / 'The Standard Model'

In 1999, Riesenhuber and Poggio presented a landmark paper describing the fundamental elements of object recognition models in the visual system (Riesenhuber and Poggio 1999). These principles were exemplified in a computational model, HMAX, also known as the *standard model*. It was labelled *standard* as it attempts to consolidate in a single model many of the widely accepted facts and observations in the primate visual system (more specifically, the ventral path). It has subsequently been employed to simulate other phenomena such as attention (Walther and Koch 2007), biological motion (Giese and Poggio 2003) and learning using spike-time dependent plasticity (STDP) (Masquelier and Thorpe 2010). It is also the backbone of the architecture used for the model in this thesis, and for that reason it will be described in greater detail in this section.

The HMAX model attempts to reproduce activity and functionality observed along the ventral visual pathway, comprising areas V1, V2, V4 and IT. The model is based upon widely accepted basic principles such as the hierarchical arrangement of these areas, with a progressive increase in receptive field size and complexity of preferred stimuli, as well as a gradual build-up of invariance to position and scale as we move further up the hierarchy. These concepts have been described in Section 2.1.

Several versions of the model have been published, although they all share the same underlying structure. It usually comprises three different levels representing V1, V2/V4 and IT, which are subdivided into two layers, simple and complex. Figure 2.4 shows a schematic representation of the HMAX model including the different types of units and operations, and the mapping onto the visual cortex.

Two operations are performed in alternating layers of the hierarchy: the invariance operation, which occurs between layers of the same level (e.g. from S1 to C1); and the selectivity operation implemented between layers of different levels (e.g. from C1 to S2).

Invariance is implemented by applying the *max* function over a set of afferents selective to the same feature but with slightly different positions and sizes. Thus, the response of a complex

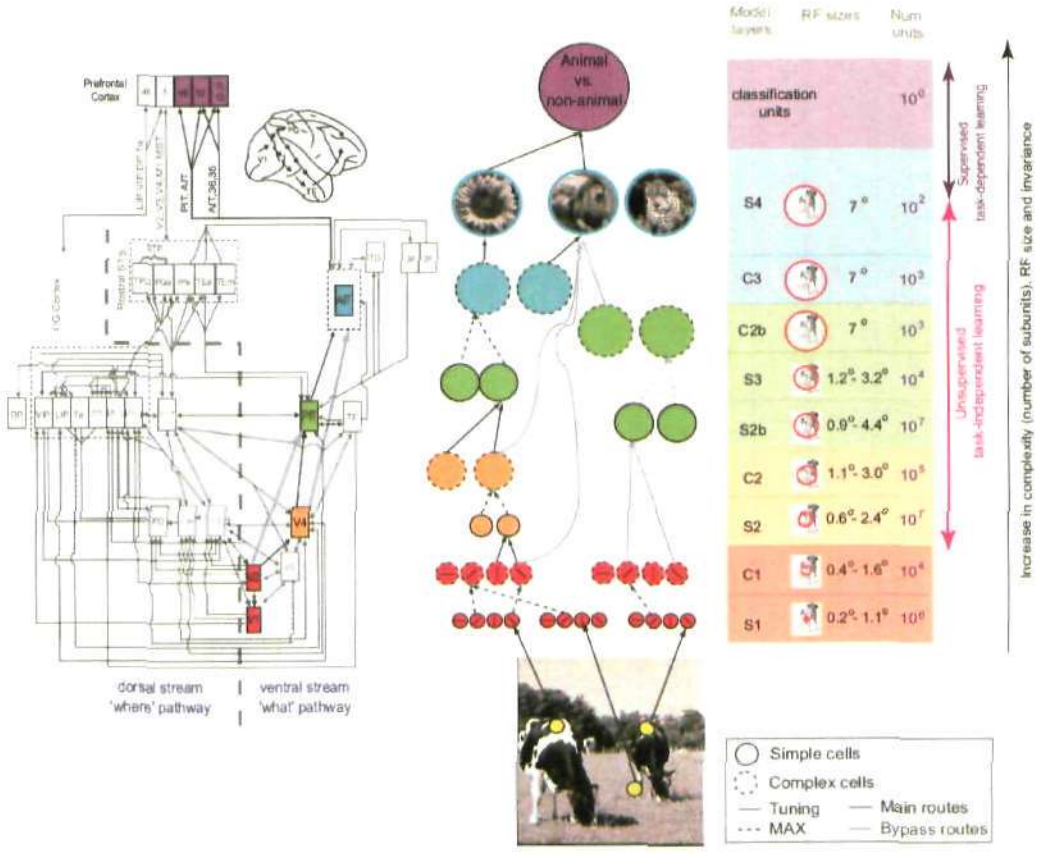


Figure 2.4: Schematic representation of the HMAX model (right) with tentative mapping over the ventral stream in the primate visual cortex (left). The model attempts to reproduce activity and functionality observed along the ventral visual pathway, comprising areas V1, V2, V4 and IT. The model is based upon widely accepted basic principles such as the hierarchical arrangement of these areas, with a progressive increase in receptive field size and complexity of preferred stimuli, as well as a gradual build-up of invariance to position and scale as we move further up the hierarchy. Two operations are performed in alternating layers of the hierarchy: the invariance operation (the *max* function over a set of afferents selective to the same feature) which occurs between layers of the same level , e.g. from S1 to C1 (dotted circles and arrows); and the selectivity operation (a template-matching operation over a set of afferents tuned to different features) implemented between layers of different levels , e.g. from C1 to S2 (plain circles and arrows). The main route to IT is denoted with black arrows, and the bypass route is denoted with blue arrows. Colours indicate the correspondence between model layers and cortical areas. The table (right) provides a summary of the main properties of the units at the different levels of the model (Serre et al. 2007b).

unit will be equivalent to the response of the afferent simple unit with the highest value. If any of the simple units within the complex unit's spatial pooling range is activated, then the complex unit will also emit an equivalent response. This means complex units achieve a certain degree of invariance to spatial translation and scale.

Selectivity is generated by a template-matching operation over a set of afferents tuned to different features, implemented as a Radial Basis Function network (Bishop 1995). First, a dictionary of features or prototypes is learned. Each prototype represents a specific response configuration of the afferent complex units from the level below, feeding into the simple unit in the level above. Each simple unit is then tuned to a specific feature of the dictionary, eliciting the maximum response when the input stimuli in the spatial region covered by the unit matches the learned feature. The response is determined by a Gaussian tuning function which provides a similarity measure between the input and the prototype. The mathematical formulation for both the selectivity and invariance operations is described in Section 4.4.

With respect to the implementation of the top level, in the first proposed model (Riesenhuber and Poggio 1999) this was described as a set of view-tuned units connected to the output of the C2 layer. The weights were set so that the center of the Gaussian associated with each view-tuned unit corresponded to a specific view of an input image. More recent versions have employed C2 features as the input to a linear support vector machine (Serre et al. 2005b, 2007c), or have implemented an additional unsupervised S3/C3 level analogous to the intermediate level (Serre et al. 2005a). In one particular implementation the model was extended to include an additional supervised S4 level trained for a categorization task, possibly corresponding to categorization units in prefrontal cortex (Serre et al. 2007b,a). A further extension, consisting of two extra sublevels S2b and C2b, has enabled some of the models to account for bypass routes, such as direct projections from V2 to IT which bypass V4 (Serre et al. 2005a, 2007b,a).

Learning in the model takes place at the top level in a supervised way, while at the intermediate levels the feature prototypes are learned in an unsupervised manner. The model implements developmental-like learning, such that units store the synaptic weights of the current pattern of activity from its afferent inputs, in response to the part of image that falls within its receptive

field. The model simulates the temporal variation in the input images (motion) during learning by generalizing the selectivity of the unit to units in the same feature map across scales and positions. Furthermore, a recent study showed how spike-time dependent plasticity could be used to generate the selectivity weights between layers C1 and S2 (Masquelier and Thorpe 2007).

On the other hand, learning is not explicitly implemented at the bottom level, as the filter responses are hard-wired. These were initially characterized as derivative of Gaussian functions (Riesenhuber and Poggio 1999). In later versions (Serre et al. 2007c,b,a, 2005a) they were replaced by Gabor functions and the receptive field size and pooling parameters of the lower and intermediate levels were more closely tuned to anatomical and physiological data (Serre and Riesenhuber 2004).

It is important to emphasize the relation between the HMAX model and neurophysiology. With respect to the response of units at different levels, the Gabor filter has been shown to provide a good fit with data from cat striate cortex (Jones and Palmer 1987). Moreover, the model parameters were adjusted so that the tuning profiles of S1 units match those of V1 parafoveal simple cells in monkeys. Further adjustment of the pooling parameters resulted in the tuning properties of S1 and C1 units being in good agreement with physiological data on simple and complex cells. This provides realistic values for the receptive field size, spatial frequency and orientation bandwith of the lower level model units (Serre and Riesenhuber 2004). Nonetheless, it is still a very simplified account of V1 neuron properties. For example, the model doesn't make any distinction between the parvocellular and magnocellular streams and ignores V1 neurons concentrated in layer 4C beta which lack orientation specificity.

Similarly, the S2-C2 hierarchy was shown to produce both selectivity and invariance that matches observed responses in V4 (Cadieu et al. 2007). Regarding the top-level units in the model, these present bigger receptive fields and are tuned to complex composite invariant features, which are consistent with the so-called view-tuned cells present in the higher levels of the ventral pathway, such as the IT cortex (Hung et al. 2005, Serre et al. 2007a,c).

The two operations performed in the model, *max* for invariance and Gaussian-tuning for selec-

tivity, stem from the original Hubel and Wiesel proposal (Hubel and Wiesel 1965), and have been supported by posterior physiological findings. Neurons in area V4 in the primate (Gawne and Martin 2002) and complex cells in the cat visual cortex (Lampl et al. 2004) have both been found to show responses that can be predicted relatively well by the *max* operation. In the latter study, when optimal and non-optimal bars were presented simultaneously, the response of the complex cells closely resembled the response when the optimal stimulus was presented alone. A recent study (Masquelier et al. 2007) demonstrates the plausibility of this mechanism, by learning complex cell invariance from natural videos. For the selectivity operation, a normalized dot product operation followed by a sigmoid function has been suggested as a biologically plausible implementation (Serre et al. 2005a, 2007c).

Although HMAX is a relatively abstract model, several attempts have been made to show its validity at a lower level of description. The *max* operation, which achieves invariance, has been shown to be implementable by different biologically plausible circuits, the most likely being the cortical microcircuits consisting of lateral and recurrent inhibition (Yu et al. 2002). Interestingly, a similar study (Kouh and Poggio 2008) extended the previous results showing how the two distinct neural operations, selectivity and invariance, were approximated by the same canonical circuit, involving divisive normalization and nonlinear operations. The circuit was based on neurophysiological data suggesting the existence of a basic cortical structure similar within and across different functional areas. At the biophysical level of description, Knoblich et al. (2007) proposed a detailed model that could approximate the HMAX operations, based on standard spiking and synaptic mechanisms found in the visual and barrel cortices. Their model was shown to implement both the invariance and tuning operations, satisfying the timing and accuracy constraints required to perform object recognition in a biologically plausible manner.

Taken as a whole the HMAX model provides useful insights into how the selectivity and invariance properties observed along the ventral path can be gradually built. It is grounded on widely accepted neurophysiological principles, such as a hierarchical increase in receptive field size and complexity. The model provides a relatively good fit to V1 cells' tuning parameters and

shows high level responses that are consistent with our current knowledge of extrastriate cortex functionality. These responses reproduce V4 shape selectivity distributions and predict human performance during a rapid categorization task.

The model also has several serious limitations. Firstly, the framework relies entirely on a feed-forward architecture, ignoring many connections which are known to exist along the visual pathways. Both long-range horizontal and feedback connections are likely to play an important role in modulating and integrating information across cortical regions. To what degree these are involved in early stages of immediate object recognition is still an open question (Hochstein and Ahissar 2002, Lee 2003). Secondly, at present the model only provides a static account of the recognition process, i.e. each unit produces a single response for a given input image. This clearly doesn't capture the complexity and dynamics of neural computations in cortex, and omits challenging aspects, such as the temporal evolution of responses and the interplay between excitation and inhibition to achieve stability. Thirdly, learning in the model occurs off-line during an initial training stage, and assumes a set of hard-wired features in the lowest level (S1). The model could be improved by adding online learning and adaptation mechanisms, such as Hebbian or spike-time-dependent plasticity, and possibly learning S1 tuning profiles in an unsupervised manner.

2.1.2.2 Neocognitron

Previous to HMAX, Fukushima had proposed the Neocognitron model (Fukushima 1988) which, due to its functional similarities, can be considered one of HMAX's predecessors. The model consists of a hierarchical network that can be trained to perform object recognition based on the similarity in shape between patterns. Recognition is not affected by deformation, changes in size or shifts in the position, thus resembling the invariance properties captured by HMAX and present in the visual system. Similarly, each level of the network consists of simple cells, which extract the features; and a layer of complex cells, which allow for the invariance properties by pooling over a group of simple cells.

The main attribute that differentiates the previous two models is the *max* operation introduced in HMAX complex layers as a new pooling mechanism. This allows HMAX to isolate the

response from the feature of interest from irrelevant background activity, increasing the recognition robustness to translations, scaling and clutter. Furthermore, the Neocognitron places a stronger focus on pattern recognition and less emphasis on capturing the anatomical and physiological constraints imposed by the visual system.

2.1.2.3 Fragment-based hierarchies

Ullman (2007) proposes representing objects within a class as a hierarchy of common image fragments. These fragments are extracted from a training set of images based on criteria which maximize the mutual information of fragments, then used as building blocks for a variety of objects belonging to a common class. The fragments are then divided into different types within each class of object, e.g. eyes, nose, mouth etc. for face recognition. During classification, the algorithm then selects the fragment of each type closest to the visual input following a bottom-up approach. Evidence from all detected fragments is combined probabilistically to reach a final decision. By using overlapping features with different sizes and spatial resolutions, the model is able to achieve a certain degree of position invariance. Later versions of the model also include top-down segmentation processes, which are beyond the scope of this chapter.

The fragment-based method introduces several novelties in relation to previous feature-based approaches: object fragments are class specific, are organized into fragment types with varying degrees of complexity, and employ new learning methods to extract the most informative fragments. However, the model is derived from computer vision approaches, hence relating to the visual system only at a very abstract level. Some basic principles of hierarchical object recognition are captured and the author puts forward psychophysical and physiological evidence suggestive of class specific features emerging in the visual system during category learning. Feature tuning is not based on physiological data (e.g. V1 features are richer than the standard model suggests), connectivity is not derived from cortical anatomy but from the image fragmentation process, and a biologically plausible implementation of the model operations has not been demonstrated.

2.1.2.4 Visnet

The model (Wallis and Rolls 1997, Rolls and Milward 2000) comprises a series of competitive convergent networks organized in four hierarchical layers. The networks allows neurons to learn combinations of features that occur in a given spatial arrangement. The feedforward connections converging on a cell at a given layer originate from a small region of the preceding layer, hence allowing an increase of the receptive field size through the layers. Most importantly, a modified Hebb-like learning rule called the trace rule, allows neurons to achieve invariance to several transformations, analogously to IT cortex neurons.

The trace learning rule incorporates a decaying trace of each cell's previous activity, hence adapting synaptic weights according not only to current firing rates, but also to the firing rates elicited by recently seen stimuli. By studying natural image statistics, it is easy to conclude that slowly changing input over a short period of time is likely to belong to the same object. Therefore, by presenting sequences of gradually transforming objects, the cells in the network learn to respond similarly to all the natural transformations of an object.

In contrast to the Neocognitron and HMAX, which employ different mechanisms to attain invariance and selectivity, Visnet manages to resolve both using an homogeneous architecture. This is achieved by implementing the trace rule, a biologically plausible self-organizing competitive learning method.

One of the main limitations of the model is that it has been trained and tested with relatively few stimuli, compared to other models such as HMAX. The later version of the model, Visnet2 (Rolls and Milward 2000), increased the number of stimuli in the dataset, although it was still *limited to images of faces and only invariance to translation (faces at different locations) was* tested.

2.1.2.5 Slow Feature Analysis

This method, introduced by Wiskott, allows the model to extract a set of invariant or slow-varying features from temporally varying signals. It can be used to build simple models of object recognition in the visual cortex, by constructing a hierarchical network of these slow-

feature analysis modules. Results show this type of network can learn invariance to translation, size, rotation and contrast, achieving good generalization to new objects even using only a small training dataset (Wiskott and Sejnowski 2002, Mathias et al. 2008).

The slow feature principle is closely related to the trace rule employed in the Visnet model (Rolls and Milward 2000) previously described. In contrast, the main advantage of slow feature analysis is that it is not limited to extracting a single invariant representation, i.e. object identity, but also maintains a structured representation of other parameters such as object position, rotation angles and lighting direction.

2.2 High-level feedback

The previous section acts as an introduction to the visual system and in particular to object recognition. This provides the context to discuss the role of high-level feedback in perception, exposing many of the phenomena which remain unexplained and challenging some of the existing classical concepts. To avoid misinterpretation, we define feedback as activity originating in a high-level region targeting a lower-level region, which therefore excludes intralevel interlaminar activity.

2.2.1 Experimental evidence

2.2.1.1 Anatomical perspective

From the anatomical point of view, feedback connections extensively outnumber feedforward sensory pathways (Felleman and Van Essen 1991, Macknik and Martinez-Conde 2007). The great majority of connections between regions shown in Figure 2.1a are reciprocal, which provides lower areas with massive feedback from higher cortical areas. For example, cat LGN interneurons receive 25% of their inputs from the retina, while 37% come from cortex; for LGN relay cells, the corresponding percentages are 12% and 58% (Montero 1991). The same is true for thalamic relay nucleus (TRN), which mediates the transfer of information to the cortex, where the largest anatomical projection is from connections of cortical feedback and not the ascending collaterals of relay cells (Sillito and Jones 2008). For LGN relay cells it is generally believed that feedback exerts a modulatory influence, whereas cortical feedback to TRN is more

likely to drive cell responses. Synchronized feedback from layer 6 cells is likely to exert rapid and very strong effects on TRN cell responses. Sillito and Jones (2008) argue this might be a consequence of the greater proportion of AMPA receptors that are found on TRN cells vs. relay cells.

In the primate area V1 it has been estimated that less than 2% of the synaptic input to layer 4Ca originated from the magnocellular layers of the LGN, and between 4% and 9% of synaptic input to layer 4Cb originated from the parvocellular layers of the LGN Peters et al. (1994). Despite these astonishing facts, which suggest feedback must have an important role in cortical function, feedback connections have been largely ignored, or considered to play a minor function, until recent years.

It has been suggested that the massive feedback versus feedforward connectivity ratio does not necessary imply feedback connections are functionally more relevant. Macknik and Martinez-Conde (2007) argue that because higher visual areas are more selective than lower visual areas, they require larger connectivity to fill the entire lower-level feature space. Otherwise they would impose higher-level receptive field properties on the lower level. For example, for each unoriented thalamocortical feedforward projection, there must be many differently oriented corticothalamic feedback connections to represent the entire orientation space at each retinotopic location. Otherwise, LGN receptive fields would show a substantial orientation bias. This suggests the relative large number of feedback connections would be necessary even if their functional role was limited in comparison to that of feedforward connections, e.g. if feedback was limited to attentional modulation.

Macknik and Martinez-Conde (2009) further argue for a weaker and more modulatory role for feedback connections than initially suggested by anatomical considerations, based on the following three arguments. Firstly, whereas the thalamocortical feedforward connections may be potentially active irrespective of stimulus orientation at a given time, only a small fraction of the corticothalamic feedback connections (e.g. a specific orientation) will be functionally active. Secondly, the *no-strong-loops* hypothesis states that neural networks can have feedback connections that form loops, but they will only work if the excitatory feedback is not too strong.

Thirdly, physiological findings, some of which are described further on in this section, indicate feedback plays a modulatory rather than a driving role.

A number of clarifications and remarks on Macknik's theory are now described. Firstly, it is important to make clear that the feedforward projection from an unoriented LGN cell to several V1 oriented neurons consists of a single axonal fibre which branches at the end to make the different synaptic connections. However, each V1 neuron requires an individual axonal fibre to feedback to the original LGN neuron. Thus, although the number of feedforward and feedback synaptic connections is equivalent, a larger number of feedback axonal fibres is required. The explanatory diagram included in Figure 81.3 in Macknik and Martinez-Conde (2009) shows a one-to-one relationship between feedforward and feedback connections (depicted as arrows), which contrasts with the one-to-many relationship described in the text, and may lead to confusion.

Furthermore, it is not clear whether Macknik's principle generalizes to higher extrastriate areas, presumably with a larger feature space at each location and consequently a more sparsely distributed connectivity pattern. For instance, it seems unlikely that a neuron coding for a specific orientation in V1 receives feedback from all the complex features coded in V4 at that location.

Overall, Macknik's claims seem reasonable and provide an explanation for the large ratio of feedback to feedforward projections in the visual system. Nonetheless, after taking into account this consideration, the effective connectivity ratio is still significant (one-to-one), thus still constituting an argument for a potentially strong functional role for feedback connections. With respect to the relatively weak modulatory effects attributed to feedback, it must be noted that the modulatory strength might be dependent on the characteristics and context of the input stimuli. For example, the weight of top-down feedback might be stronger for more ambiguous images.

2.2.1.2 Physiological evidence

In line with the functional role of feedback suggested by anatomical considerations, from a physiological perspective, there exists abundant evidence showing the modulatory effects of high-level feedback on lower levels. The most simple example comes from cortical areas feed-

ing back to LGN, which sharpens the synchronization of spikes driven by contours precisely aligned over the LGN RF, leading to a sharpening of the orientation tuning curves of V1 cells (Andolina et al. 2007, Sillito et al. 2006). Similarly, feedback from area V2 modulating V1 has been extensively reported. For example, inactivation experiments of area V2 (Angelucci et al. 2002, Angelucci and Bullier 2003) resulted in a decrease in V1 neurons' response. In general, inactivation of area V2 leads to a reduction of V1 response, although in some cases enhancement has also been observed, specially in the regions surrounding the classical receptive field (Macknik and Martinez-Conde 2009). The orientation selectivity and other functional characteristics of most V1 neurons remained unaltered.

Further evidence suggests the involvement of V2 in mediating end-stopping in V1, a phenomenon whereby cells decrease their response when the stimulus size exceeds the classical receptive field. The experiment showed how cells in the infragranular layers of V1 lose the end-stopping property when the supragranular layer, which receives feedback from V2, is inactivated (Bolz and Gilbert 1986). Similarly, the temporal evolution of illusory contour formation, as well as other properties described in detail in Section 2.3, are suggestive of feedback from V2 being involved in illusory contour formation in V1 (Murray et al. 2002, Seghier and Vuilleumier 2006, Lee and Nguyen 2001).

Higher processing areas associated with object recognition, such as the postero-temporal visual cortex in cats, were also reported to influence the response of V1 neurons. Inactivation of this high-level region (by cooling) generally reduced the response magnitude of V1 neurons (Figure 2.6a), and provoked substantial changes in their orientation tuning widths or direction selectivity indices (Huang et al. 2007). Analogously, the lateral occipital complex (LOC) region, associated with object recognition in humans, was also found to have an effect on lower cortical levels (Murray et al. 2004, Williams et al. 2008, Fang et al. 2008). Furthermore, during mental imagery, natural object categories have been robustly readout from the LOC region (Reddy et al. 2010) and retinotopically organized activations have been observed in early visual areas (Slotnick et al. 2005). This suggests that during mental imagery, in the absence of bottom-up input, cortical feedback projections can selectively activate patterns of neural activity (Reddy

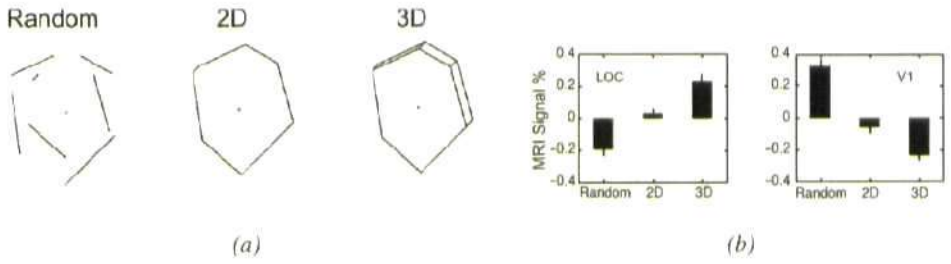


Figure 2.5: a) Stimuli with similar spatial properties but increasing organizational complexity: random lines, 2D shape and 3D shape. b) fMRI percent signal change in V1 and LOC regions for the three conditions. Percent signal change is from the mean activation across all three conditions. Although the stimulated input regions were very similar, activity in V1 showed a reduction for the 2D shape and further reduction for the 3D shape stimulus. The LOC area exhibited the opposite pattern, an increase in activity proportional to the complexity of the input figure. The author suggests activity in lower areas is reduced when a simpler explanation of the stimulus can be represented in higher areas Murray et al. (2004).

et al. 2010, Ishai 2010).

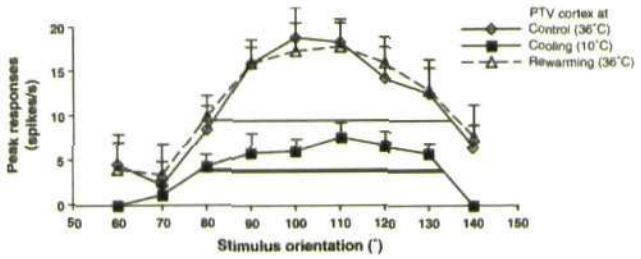
An illustrative example comes from Murray et al. (2004), who recorded V1 and LOC fMRI activity in response to three different stimuli: random lines, lines of similar length arranged to form a 2D shape, and similar lines to form a 3D shape. Although the stimulated input regions were very similar, activity in V1 showed a reduction when the 2D shape instead of the random lines was presented, and further reduction when the 3D shape stimulus was employed. Interestingly, the opposite pattern was observed in the LOC area, i.e. it exhibited an increase in activity proportional to the complexity of the input figure (Figure 2.5). The author suggests activity in lower areas is reduced when a simpler explanation of the stimulus can be represented in higher areas (linked to perceptual grouping), which implies feedback modulatory effects between LOC and V1.

Experiments involving feedback in the visual system have placed a strong focus on area MT, believed to be involved in motion processing. Although it is not the focus of this thesis, it serves to illustrate the ubiquitous presence of feedback effects in visual processing. Proof that high-level motion processing modifies the lower level's response has been shown by inactivating MT (Hupe et al. 2001, Galuske et al. 2002); artificially stimulating MT (Sillito et al. 2006); statistical coupling of V1 and MT in the context of apparent motion (Sterzer et al. 2006);

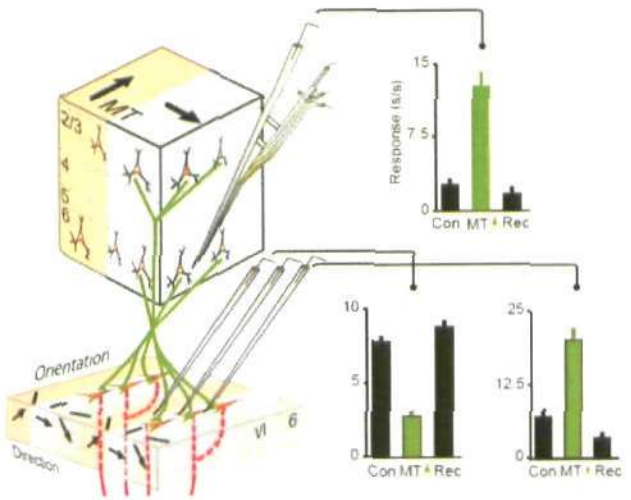
and comparing the local V1 response to coherent vs. incoherent global motion (Harrison et al. 2007). Figure 2.6b illustrates the differential results observed in V1 cells' response during artificial stimulation of the reciprocally connected region MT. In general, V1 simple cells showed an increase in response, while non-oriented cells exhibited a reduction in activity, suggesting area MT can potentially shape the response properties of V1 cells (Sillito et al. 2006). More recently, a revealing study showed how V1 cells' response was reduced when their onset or motion direction could be predicted by surrounding illusory motion (Alink et al. 2010). The surrounding stimuli were well outside the classical receptive field, suggesting the involvement of the visual motion area MT.

An important question to ask is whether horizontal connections, rather than feedback connections, could be responsible for the described contextual effects. Theoretically, since receptive field size increases and magnification factor decreases with cortical distance from V1, feedback connections from the extrastriate cortex can convey information to a V1 neuron from much larger regions of visual field than the V1 neuron can access via horizontal connections. This matter was addressed in studies by Angelucci and Bullier (2003) and Angelucci et al. (2002) who compared the spatiotemporal properties of the two potential candidates: feedback and horizontal connections (Figure 2.7). They used injections of sensitive bidirectional tracers in V1 to estimate the extent (measured in visual field degrees) of feedback connections from areas V2, V3 and MT. Results confirmed that feedback spatial properties provided a substrate for all surround modulations, including those originating from the distal surround (over 13°). Additionally, feedback terminations in V1 are retinotopically and functionally organized, for example according to orientation preference (Angelucci and Bullier 2003, Macknik and Martinez-Conde 2007). This makes them suitable for explaining the modulatory surround effects observed experimentally.

With respect to the temporal properties, experimental results have shown both feedback and feedforward pathways are made of fast-conducting axons with a median velocity of 3.5 m/s (Girard et al. 2001). The speed of connections refer to effects mediated via the dorsal path, whereas effects mediated by the ventral/parvocellular path are likely to be slower. Nonetheless,



(a)



(b)

Figure 2.6: a) Evidence showing the inactivation of postero-temporal visual (PTV) cortex reduces V1 response. The graph shows the peak response rates of the same cell obtained for different orientations before cooling, during cooling and ~ 30 mins after rewarming PTV cortex. Note that despite a dramatic reduction in the magnitude of responses during cooling, there is an excellent recovery of the magnitude of responses after rewarming the PTV cortex (Huang et al. 2007). b) Enhancement of MT modulates V1 response to visual stimulation. The visual response magnitude is enhanced in the MT cell (top graph) via a small iontophoretic current of the GABAB receptor antagonist CGP. Enhancement of MT feedback evoked either an increase (bottom-right graph) or decrease (bottom-left graph) of the response in V1 cells to the same stimulus driving the MT cell. All graphs show the magnitude for the control condition (black bar labeled *Con*), during MT stimulation (green bar labeled *MT*) and after recovery (black bar labeled *Rec*) (Sillito et al. 2006).

this suggests that, despite large differences in axonal lengths, initial processing stages can be considered to act within a common temporal window. This is consistent with the fast decrease of response observed (within the first 10 ms bin) on areas V1, V2, and V3 when inactivating area MT (Hupe et al. 2001).

Horizontal connections are intralaminar, although also usually reciprocal and linking cortical points of similar functional properties. Typically they do not drive target neurons but elicit subthreshold modulatory responses. However Angelucci and Bullier (2003) demonstrated that the monosynaptic range of horizontal connections cannot account for surround effects extending beyond the classical receptive field (approx. 2°). Although polysynaptic circuits could in principle underlie these long-distance effects, the slow conduction velocity of horizontal connections makes it highly unlikely. Girard et al. (2001) showed horizontal connections have a speed of approx. 0.1 to 0.2 m/s, which is more than ten times slower than feedback connections. For example, the effects of surround stimuli located 13° away (equivalent to approx. 1 cm of V1 cortical surface), at a speed of 0.1 m/s, would take over 100 ms to arrive. This is inconsistent with long range effects observed during the very early stages of the response (Hupe et al. 2001).

Although this section is restricted to describing feedback within the visual cortex, there are examples present in many other areas. These include higher-level areas related to decision-making, expectations (Summerfield and Egner 2009), emotions (Sabatinelli et al. 2009) and motor-planning (Grossberg et al. 2007). One such example is the orbitofrontal cortex, which has been found to feed back to the fusiform gyrus in the temporal cortex, providing top-down facilitation during object recognition (Bar et al. 2006). Similarly, there is evidence suggesting feedback is present in other hierarchical sensory processing areas, such as the auditory system, where the superior temporal gyrus is believed to exert top-down modulating effects on the primary auditory cortex (Garrido et al. 2007). This strongly argues for considering feedback as a fundamental element in cortical processing.

2.2.2 Theoretical implications

In the past few years a great effort has been made to try to understand the experimental results relating to feedback. Recent reviews (Carandini et al. 2005, Olshausen and Field 2005) of our

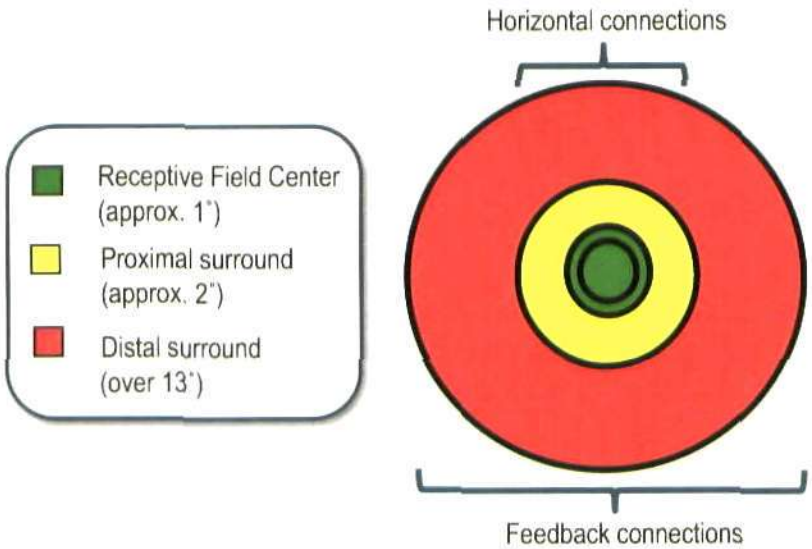


Figure 2.7: Comparison of the spatial extent of horizontal and feedback connections. The diagram schematically represents results obtained using injections of sensitive bidirectional tracers in V1 to estimate the extent in visual field coordinates of feedback connections from areas V2, V3 and MT. Note that the size of the receptive field centre will depend on eccentricity. Results confirmed that feedback spatial properties provided a substrate for all surround modulations, including those originating from the distal surround (red region). However, the monosynaptic range of horizontal connections could only account for surround effects within the classical receptive field (red region) and the proximal surround (yellow region) (Angelucci and Bullier 2003). Note diagram is not to scale.

understanding of the early visual system suggest only approximately 20% of the response of a V1 neuron is determined by conventional feedforward pathways, while the rest arises from horizontal and feedback connectivity. However, despite growing evidence, the way in which feedback operates in the brain is still far from being understood. This was highlighted in Section 2.2.1 by the lack of homogeneity and seemingly contradictory feedback effects observed experimentally, which sometimes act to enhance and other times to suppress lower levels' activity.

The problem is rooted in a wider issue, which lies at the core of neuroscience: understanding the intricate relationship between all the different regions involved, and how all these different sources of information are integrated over time and space. From this perspective, visual neural responses not only depend on the interaction between stimulus and the surrounding context (Schwartz et al. 2007), but can also be affected by other sensory modalities, attentional priors, expectations, previous experience, emotional states, or task-oriented motor plans (Gilbert and Sigman 2007, Sasaki et al. 2010). On top of this, neural responses may be involved in different processing stages which evolve over time. Feedback undoubtably plays a major role in this complex process.

Although, as has been pointed out, many factors can potentially modulate visual responses, this section focuses only on the interactions between the different visual cortical regions. It describes several theoretical approaches derived from experimental observation, dealing with spatial contextual influences (extra-classical receptive field), time-evolving processing stages at different regions, and distinct modes of processing dictated by high-level properties (Reverse Hierarchy Theory). The section concludes by discussing a related significant aspect, namely the relationship between feedback and the neural representation of conscious visual perception.

2.2.2.1 Extra-classical receptive field.

The experimental evidence presented in the previous section makes clear that V1 neurons are not only specialized for extracting local features, such as orientation, but also respond to events distant from the stimulation of their classical receptive field. One such experiment (Harrison et al. 2007) clearly illustrates this by measuring whether V1 responses are sensitive to the global

context of motion. Two different sets of stimuli were used, one moving coherently and one incoherently. In each case the stimulus inter-element distance was at least 3° apart, so from a local perspective they were all identical. We define local as being within the range of the proximal surround field, which is about 2.3°. Nevertheless, V1 responses were sensitive to the global context of motion, implying their receptive field comprises not only the proximal surround field, but a further region which is known as the extra-classical receptive field. Another remarkable study in support of this concept showed that the feedback-mediated response in the foveal retinotopic cortex contains information about objects presented in the periphery, far away from the fovea, even in the absence of foveal stimulation (Williams et al. 2008).

This shift in the traditional view of receptive field was reinforced by the study comparing horizontal to feedback connections (Angelucci and Bullier 2003, Angelucci et al. 2002), described in Section 2.2.1. Anatomical and physiological data indicated that the spatiotemporal properties of feedback connections from higher levels provided a plausible substrate for all observed extra-classical receptive field effects. Horizontal connections could also be involved, but only in center-surround interactions within the proximal surround range.

2.2.2.2 Integrated model of visual processing.

One main implication that can be derived from the existence of high-level feedback is that information doesn't necessarily have to be processed serially through successive cortical areas. Instead, multiple areas can carry out simultaneous computations, which evolve over time as successively higher cortical regions become involved in the process (Lee 2003, Lee et al. 1998). For example, evidence shows that the early part of V1 neuronal response is correlated with the orientation of local features, while the later response is correlated with higher order contextual processing. It has been suggested that V1 could potentially take an active part in all the different processing stages usually attributed to higher levels, such as the representation of surface shapes or object saliency (Hochstein and Ahissar 2002, Lee 2003, Bullier 2001, Lee et al. 1998).

This idea is consistent with conceptualizing V1 as an *active blackboard* (Bullier 2001) or high-resolution buffer (Lee 2003). Higher cortical areas feed back global and contextual information to complete or update the high-resolution detailed representation maintained at the lower levels,

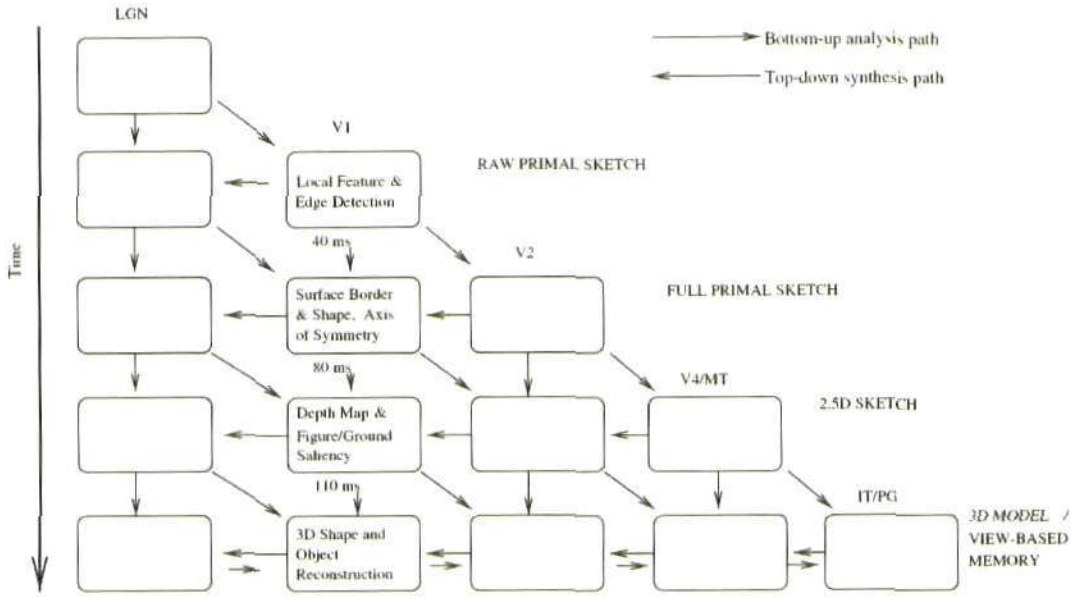


Figure 2.8: Evolution of the computations performed in the different visual areas over time. Each column represents the involvement of a particular region in different computations over time. Each row represents parallel computations, proposed by Marr (1982), across the multiple areas. As time progresses, the activity in V1 should reflect the involvement in increasingly complex computations resulting from recurrent feedback from higher cortical areas (Lee 2003).

in congruence with the extra-classical receptive field idea. This information is then propagated up the hierarchy again providing a new level of analysis which can be employed for the next stage of computation. As time progresses, the activity in V1 should reflect the involvement in increasingly complex computations resulting from recurrent feedback from higher cortical areas. A representation of the suggested temporal evolution of the functions carried out by the different areas involved is illustrated in Figure 2.8. This challenges the classical serial feedforward model depicted in Section 2.1.1.

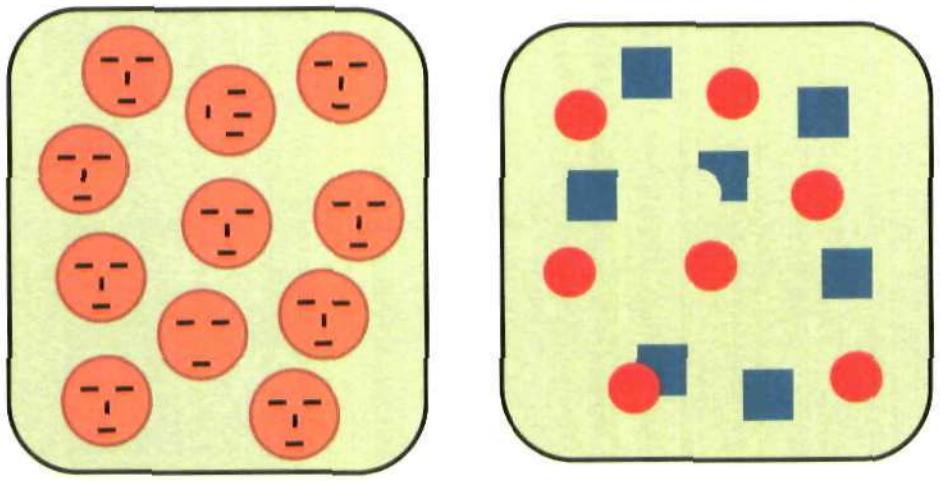
2.2.2.3 Reverse Hierarchy Theory (RHT).

This theory formulated by Hochstein and Ahissar (2002), proposes an interesting theoretical point of view of the role of feedback connections in visual perception. Explicit visual perception is hypothesized to occur for the first time at the higher cortical levels, after receiving

detailed information from low-level areas. The initial bottom-up feedforward processing is considered to be only implicit and not directly available to conscious perception. This type of representation, denoted as *vision at a glance*, is obtained by generalizing low-level information over space, size and viewpoint, leading to high-level neurons which indicate the presence of basic categories or objects but not their precise parameters. Later, during *vision with scrutiny*, feedback connections gradually incorporate the lower-level details into the explicit perceptual representation. This includes features such as the precise location, retinal size, color, or component motion, which are only available in the lower cortical areas with smaller receptive fields, and were lost in the neurons with larger RFs. This concept is consonant with the high-resolution buffer and integrated model of visual processing previously described.

Vision at a glance has been associated with a type of search called feature search, characterized by the amazing ability in humans to rapidly capture object categories. Although low-level areas were thought to be responsible for feature search, several arguments suggest this ability, which has been related to fast pop-out mechanisms (approx. 100 ms from stimulus onset), actually reflects high-level cortical activity. Feature search works for a vast range of spatial scales, sizes and inter-element distances, including values which are greater than the small low-level receptive field sizes. These parameters are consistent with the high-level large receptive fields which reflect spread attention and lead to position and size invariant feature detection. Furthermore, the fast pop-out effect observed is usually related to high-level features, such as depth from shading, 3D shapes or facial expressions. An example is shown in Figure 2.9a where the non-face object immediately pops-out from the rest of the similar line drawings. Another example is depicted in Figure 2.9b where an incomplete square rapidly pops out while an identical shape is interpreted as an occluded square due to amodal completion, a feature of implicit high-level processing.

On the other hand, *vision with scrutiny* is associated with serial or conjunction search. This is illustrated by initial blindness to the details in a scene, which disappears after longer and repeated exposure. By focusing high-level mediated attention to different areas or objects, details from the low-level cortical representation are serially introduced. The extra-classical recep-



(a) (b)

Figure 2.9: a) Rapid high-level object categorization. The scrambled face rapidly pops out, whereas the noseless face requires a serial search. b) High-level implicit processing. The incomplete square rapidly pops out whereas an identical shape is interpreted as an occluded square (Hochstein and Ahissar 2002).

tive field approach accounts for feedback originating in high levels with large receptive fields which targets specific low-level features. An example is shown in Figure 2.9a where, although the non-facial object pops out immediately, a slower serial search mechanism is required to identify the noseless face. A similar finding reported that subjects require less time to identify target orientation than to accurately localize it. This is consistent with explicit perception later accessing low-level detailed representations, such as those encoding spatial localization.

The Reverse Hierarchy Theory also points out the initial coherence and feature binding properties of visual perception. Even for images containing ambiguous interpretations, such as illusions or bistable images, the initial explicit perception is typically of a complex coherent scene, and not of an unlinked collection of lines and coloured regions. This phenomenon can be considered a direct outcome of hierarchical perceptual organization (Ahissar et al. 2009) and the resulting receptive field of high-level object-related neurons. It is coherent with the template matching-like operation which assigns images to categories, implemented in the feedforward object recognition models described in Section 2.1.2.

The temporal evolution of activity in low-level regions is also predicted by this theory. Initial

activity generated by feedforward bottom-up implicit processing should be driven by stimuli, localized and automatic. This will give rise to the first *vision at a glance* high-level percept which might in turn activate serial search or *vision with scrutiny* mechanisms. As a consequence later activity in lower-levels will reflect feedback top-down effects such as those associated with spatial and object attention, matching the functional temporal evolution proposed by Lee (2003).

2.2.2.4 Cortical representations of conscious visual perception.

It may be inappropriate to refer to consciousness, as it is a highly controversial concept which is not well defined or understood. However, for the purpose of this section it will be interpreted as referring to the explicit visual perception previously described, sometimes also denoted as visual awareness. Which areas of the visual system are actually involved in representing the explicit or conscious visual percept? The hypothesis of a distributed representation of explicit perception is gradually gaining favour over the traditional strictly high-level cortical representation. For the authors of the RHT, explicit perception begins at high cortical levels and then proceeds in a top-down fashion, strongly influenced by attention, to gradually incorporate more detailed information from lower levels. It seems a reasonable assumption given that we are able to explicitly perceive high-resolution details which can only be accurately encoded by lower-level regions.

Supporting this argument, experiments based on perceptual rivalry conclude that it would be more appropriate to begin thinking of consciousness as "a characteristic of extended neural circuits comprising several cortical levels throughout the brain" (Wilson 2003). Along the same lines, several studies conclude unstimulated areas of V1 can represent illusory contours (Maertens et al. 2008), or the illusory perception of apparent motion (Sterzer et al. 2006), which corroborates the notion that subjective perceptual activity can be closely related to neural activity in V1. Overall, evidence seems to indicate an important role for feedback connections in mediating explicit visual perception or awareness (Leopold and Logothetis 1996).

2.2.3 Functional models of feedback

The previous sections review some of the experimental evidence which indicates feedback connections act to modulate lower levels' response. This yields several theoretical conclusions which shift the traditional bottom-up serial processing ideas towards a more integrative and dynamic approach to visual perception. It has become clear that information from different visual cortical regions needs to be combined to achieve perception, but how this happens and the exact function of feedback connections in this process is still unknown. This section describes different approaches that provide a functional interpretation of the role of feedback, including attention, biased competition, adaptive resonance, predictive coding and generative models. It is important to note that the different interpretations are not mutually exclusive and commonly have overlapping features, which means computational models often fall into more than one category. Likewise, each functional interpretation described below is likely to be consistent with a significant subset of the theoretical considerations described in the previous section.

2.2.3.1 Feedback as attention.

The visual system receives vast amounts of input information every second from light entering the retina. Attention is aimed at reducing the associated computational cost by prioritizing and consequently processing only a subset of the visual information. This subset would typically correspond to that of highest relevance in achieving the organism's goals (Summerfield and Egner 2009). The function of feedback connections would be to modulate the visual input, by enhancing or suppressing feedforward signals, in accordance with the attentional state. Attention can arise from high-level cognitive areas associated with task or motor-planning and be directed towards specific objects or locations. On the other hand, attention can also be attracted intrinsically by stimuli with strong visual salience, such as a sudden motion, which might be an indicator of imminent danger.

Another distinction which is usually made relates to the way of deploying attention. It can be broadly categorized into spatial attention, which acts as a kind of spotlight that enhances the processing at a specific location of the visual field; and feature-based attention, whereby the processing of specific features is biased in a top-down fashion in order to achieve a specific

task, such as visual search.

The different types of attention have been modelled extensively. Walther and Koch (2007) provide a comprehensive overview of existing models, and propose a unifying framework which captures most of the attentional effects. By implementing modulation functions at each processing level, their model is capable of reproducing spatial and feature based attention both in a top-down and bottom-up fashion.

Additionally, the model is capable of simulating object-based attention, which can encompass a variety of effects. These range from spatially focusing on an object to enhancing the relevant features of the target object during a search task. This is achieved by making use of the same complex features employed for feedforward recognition, during the top-down attention process. The HMAX model (Serre et al. 2007c) was extended to provide an example of feature-based attention using this principle, and results showed an increased performance over a pure bottom-up attention implementation (Walther and Koch 2007). The present thesis also provides a feedback extension of the HMAX model, which has many theoretical similarities to this approach, including the sharing of features between object recognition and top-down attention.

It has been argued that attention by itself may explain the existence of cortical feedback connections (Macknik and Martinez-Conde 2007), without requiring further complex interpretations. In fact, most of the approaches allude to some kind of attentional mechanism when describing the role of feedback. In most biased competition models, e.g. Deco and Rolls (2004), attention is taken as a synonym for feedback. For adaptive resonance approaches (Grossberg et al. 2007) attention is one of the multiple functions of feedback connections, which are also involved in learning or perceptual grouping. In contrast, Lee and Mumford (2003) argue the sophisticated machinery of feedback should not be limited to biased competition models of attention, but instead should account for more complex perceptual inference processes. But even in generative-oriented models such as Bayesian inference or predictive coding, the top-down priors or high-level predictions are sometimes referred to as a form of attentional modulation (Spratling 2008a, Chikkerur et al. 2009).

The inconsistency and disparity between the various definitions of attention might reflect the

lack of understanding of the role of top-down cortical feedback. Apparently divergent interpretations might in fact be modelling a common phenomenon, but until this has been more thoroughly characterized by experimental evidence, several semantic definitions must be allowed to co-exist.

2.2.3.2 Feedback in biased competition models.

The biased competition theory (Desimone and Duncan 1995) proposes that different visual stimuli compete to be represented by cortical activity. Competition can occur at all levels in the hierarchy and is influenced both by feedforward and feedback connections. The model is consistent with an abundant body of experimental evidence (Hupe et al. 2001, Huang et al. 2007, Murray et al. 2004) which suggest feedback enhances activity consistent with the higher-level percept (see Reynolds and Chelazzi (2004) for a recent review). At the same time biased competition appears to disagree with evidence supporting the opposite effect, i.e. that the lower level response is actually reduced when it can be predicted by higher levels (Harrison et al. 2007). A possible explanation comes from the fact that the decrease of inconsistent activity is greater than the enhancement of consistent activity leading to an overall reduction in activity (Murray et al. 2004). Further ways to reconcile biased competition with predictive coding are discussed in Section 2.2.3.

Biased competition models have managed to successfully capture top-down attentional effects including spatial and object based visual search. Deco and colleagues have proposed a model of invariant visual object recognition consisting of a hierarchy of visual cortical regions with convergent feedforward connectivity, reciprocal feedback connections and local competition within each region (Deco and Rolls 2004). The model accounts for the increased attentional modulation observed in higher cortical levels and the reduced receptive field size of IT neurons in highly cluttered images.

Along the same lines, biased competition models have also been able to replicate attentional effects in V4 and IT resulting from active visual search (Lanyon and Denham 2004), as well as realistic search scan paths and saccade behaviours (Lanyon and Denham 2009). Other models also account for motion disambiguation processes between MT and V1 (Bayerl and Neumann

2004), and perceptual grouping mechanisms (Roelfsema 2006). Furthermore, some research (Tiesinga and Buia 2009) has focused on the detailed circuitry required for biased competition to emerge in V4, and concluded it can feasibly arise as a result of feedforward projections from V1 and surround suppression mechanisms.

2.2.3.3 Feedback in Adaptive Resonance Theory (ART).

Although included in a different category, ART (Carpenter and Grossberg 1987, 1998) can be considered a type of biased competition model as it has many similar properties. A central feature in ART is the matching process that compares the bottom-up input with the stored pattern. Unlike other networks, ART encodes only the matched or 'resonant' pattern and not the actual input, as it suppresses all the portions which do not match the top-down expectation. A parameter, which specifies the minimum fraction that must remain in the matched pattern for resonance to occur, ensures that if the input is too novel or unexpected a new pattern code is learned by the system.

The LAMINART model (Grossberg 2003, Grossberg et al. 2007, Raizada and Grossberg 2003) implements the described ART mechanisms, mapping them over laminar visual cortical circuits. These mechanisms are hypothesized to occur in the neocortex to help stabilize cortical development and learning. The model employs feedforward, feedback and horizontal interactions to achieve the unification of several processes including development, learning, perceptual grouping, attention and 3D vision.

A special emphasis has been placed on modelling the detailed laminar circuits of V1 and V2 in order to achieve extra-classical receptive field effects such as perceptual grouping and attention. The authors identify a list of requirements that any successful cortical model of visual perception should accomplish. Cortical models must allow perceptual grouping to generate activity in a classical receptive field with no direct visual stimuli (as happens with illusory contours) but must prevent top-down feedback from doing the same (i.e. producing above-threshold activity on its own) in order to avoid hallucinations. However top-down feedback must be allowed to provide modulatory subthreshold activity to enhance matching incoming sensory signals. This is known as the preattentive-attentive interface problem.

A solution to the problem is offered by Grossberg and colleagues (Raizada and Grossberg 2001, Grossberg and Raizada 2000, Grossberg et al. 1997) which consists of distinguishing between top-down intracortical interlaminar preattentive feedback (a positive reinforcement loop within V1 layers) and top-down intercortical attentional feedback (from V2 to V1). Perceptual grouping starts at layer 2/3, guided by bottom-up input signals and horizontal intralaminar connections. Top-down attentional feedback and preattentive feedback cells in layer 2/3 interact by reinforcing the same feedforward selection circuits. However, attentional feedback is forced to pass through a modulatory laminar circuit before reaching layer 2/3, ensuring that it can only provide subthreshold modulatory effects but never directly drive the cell.

A later model named ARTSCAN (Fazl et al. 2009, Bhatt et al. 2007) accounts for the interaction between spatial and object attention in order to search a scene and learn the object categories which are present. The model is based on the concept of an attentional shroud, a distribution of spatial attention generated by the object's surface filling-in process. Stronger shrouds can inhibit weaker ones, leading to a winner shroud which will guide the category learning process and the deployment of object attention.

The ART approach provides a detailed theory of how laminar cortical circuits implement a wide range of learning and perception-related functions in the brain. Although it doesn't place a strong focus on high-level object categorization, the ARTSCAN model proposes a role for feedback connections in the process of integrating bottom-up and top-down object-related information. It is based on the concept of attentional shrouds, includes interactions with the *where* path, but has only been tested with relatively simple character recognition tasks.

For the purpose of this thesis, which uses relatively abstract models based on Bayesian approaches to explain object perception, the HMAX model seems a better starting point than ART, for the following reasons. Firstly, HMAX is strongly oriented to object recognition and has been tested successfully on natural images. Secondly, HMAX's more abstract nature, which, for example, does not deal with intricate laminar connections, makes it more suitable to implement large-scale Bayesian computations. Thirdly, HMAX lacks feedback connectivity, which makes it a perfect candidate to test this new approach.

However, this doesn't mean the ART model's interpretation of feedback is in disagreement with Bayesian interpretations. Grossberg et al. (1997) suggests the ART approach clarifies the current appeal of Bayesian approaches, but goes beyond this type of model. It is therefore likely that many aspects of the more abstract interpretation of feedback connectivity proposed in this thesis are compatible with Grossberg's detailed circuitry.

2.2.3.4 Feedback as predictive coding

In predictive coding each level of the hierarchical structure attempts to predict the responses of the next lower level via feedback connections. The difference between the predicted and actual input is then transmitted to higher order areas via feedforward connections, and used to correct the estimate. The predictions are made on progressively larger scale contexts, such that, if the *surround* can predict the *centre*, little response is evoked by the error-detecting neurons. In other words, when top-down predictions match incoming sensory information, the lower-level cortical areas are relatively inactive. However, when the central stimulus is isolated or difficult to predict from the surrounding context, then the top-down predictions fail, and a large response is elicited.

Although predictive coding is presented as an isolated theory in this section, it is related, to a major or minor extent, to all the previous approaches. To start with, it is in fact a specific example of a broader and more general theoretical approach termed hierarchical perceptual inference in generative models (Friston 2003, 2005, Spratling 2010). This is described in detail in Chapter 3. In fact, the Kalman filter, used to implement predictive coding in a hierarchical architecture, is a particularization of the Bayesian Belief Propagation algorithm (Kschischang et al. 2001), and is derived under the Bayesian framework by maximizing the posterior probability at each layer (Rao 1999).

Furthermore, it has recently been shown that predictive coding can be interpreted as a form of biased competition model (Spratling 2008b). Traditionally these two approaches have been considered opposite to each other, as shown in Figure 2.10 (Murray et al. 2004). The discrepancy is resolved, firstly, by taking into account the two distinct subpopulations, one encoding the current prediction, or active representation of the stimuli; the other encoding the prediction

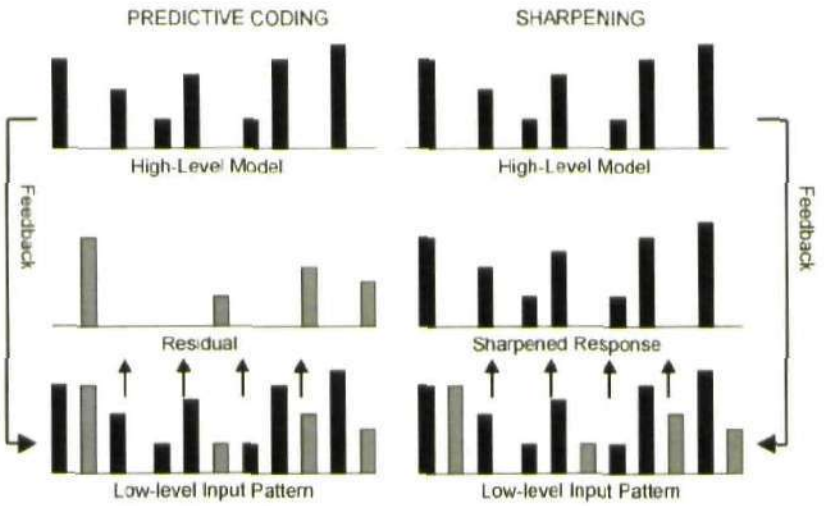


Figure 2.10: Comparison of predictive coding (left) and sharpening (right) effects in mediating response reduction in lower levels. In predictive coding, a high-level prediction of the expected input is fed back and subtracted at the input level. What is sent forward is the difference between the expected value and the actual input. With sharpening (present in biased competition and adaptive resonance models), the same high-level prediction is fed back but is instead used to enhance those aspects of the input that are consistent with the high-level percept and reduce all other aspects. The result, in both cases, can be a reduction in activity. (Murray et al. 2004).

error. While the error population will show reduction with high-level feedback, the prediction population may show enhancement. Predictive coding theories can be misleading, as they place a stronger focus on the error-detecting nodes and consequently under-emphasize or omit prediction nodes. A second requisite to reconcile biased competition and predictive coding theories regards the connectivity of feedback. In most biased competition models, nodes at each level compete by inhibiting the output of neighbouring nodes, while feedback in predictive coding typically acts on the level below. Therefore, the biased competition model that was shown to be mathematically equivalent to predictive coding (Spratling 2008b), required an alternative implementation that suppressed the inputs of neighbouring nodes.

It is not therefore surprising that predictive coding models are also compatible with theories of feedback as attention. Both Rao (2005), by extending his original model, and Spratling (2008a), using the previously described architecture, demonstrated that predictive coding could account for spatial and feature attentional effects. Furthermore, Spratling (2008a) hypothe-

sizes that perceptual grouping results from collinear facilitation, and demonstrates this using a detailed physiological model of the interaction between horizontal and feedback connections. These interactions are proposed to occur in the dendritic trees of pyramidal cells (De Meyer and Spratling 2009).

Nonetheless, as suggested by a recent study (Summerfield and Egner 2009), predictive coding might be more related to the effects of expectation, which facilitates visual perception by constraining the interpretation space based on prior information, than to attention. Although behaviourally both attention and expectation can have similar effects, they might exert opposing influences on the responses of the neural populations involved, i.e. expectation might reduce the response while attention might enhance it. However, this distinction at the neural level is only hypothetical and remains to be substantiated by further experimental and modelling work on this research topic, which is still at a very early stage.

Independently of its relationship with other theoretical approaches, predictive coding has been outstandingly successful in explaining feedback experimental results. It is also consistent with the growing collection of evidence showing that lower level responses are inversely correlated to stimulus predictability (Alink et al. 2010, Murray et al. 2004, Sterzer et al. 2006, Harrison et al. 2007, Rao and Ballard 1999). Furthermore, existing computational models can account for several well-known phenomena, such as repetition suppression (Friston et al. 2006), biphasic responses in LGN (Jehee and Ballard 2009), object segmentation (Rao and Ballard 1997, 2005); and a wide range of classical and extra-classical receptive field effects, including receptive field tuning properties, surround suppression and facilitation (Spratling 2010), and end-stopping effects (Rao and Ballard 1997, Rao 1999).

Note that predictive coding might appear incompatible with the observation that feedback acts to enhance activity consistent with the high-level percept, supported by evidence showing the response in V1 is reduced when higher areas are inactivated (e.g. by freezing MT) (Hupe et al. 2001, Huang et al. 2007, Galuske et al. 2002, Angelucci and Bullier 2003). However, as previously mentioned, this might be the result of over-emphasizing the error-detecting population over the prediction population (Spratling 2008b, Friston 2005).

2.2.3.5 Feedback as Bayesian inference in generative models

The perceptual inference framework (Lee and Mumford 2003, Friston and Kiebel 2009) provides an integrative approach that accommodates feedback as attention (Chikkerur et al. 2009), biased competition (Spratling 2010) and predictive coding (Friston 2005). It constitutes the focus of this thesis and is therefore described in detail in Chapter 3.

2.2.3.6 Key questions

Understanding the role of feedback connections in visual perception still poses many challenges which need to be resolved. The following list includes several key questions which remain unanswered.

- Is there evidence in cortex of two distinct populations, one coding for the prediction (or active representation) and one coding the prediction error? In which case, does the former account for feedback sharpening effects, and the latter for feedback response reduction effects?

- What is the specific role of feedback during a) the learning stage and b) the subsequent adaptation of visual features ?

- Can the different attentional effects be understood as part of a more integrative theory such as biased competition or predictive coding?

- What are the neural mechanisms that allow the integration and adequate weighting of the different sources of information (e.g. bottom-up stimuli, top-down feedback from several regions, horizontal connections) ?

- Are feedback effects limited to a subthreshold modulatory role or can they be understood in some cases as the driving signal? Given that Anderson and Martin (2006) demonstrates both feedforward and feedback synaptic connections can be considered to have *driving* properties, can the temporal dynamics account for the functional asymmetries observed?

- Does feedback allow V1 to be progressively involved in more complex computations,

typically attributed to high-level regions, as suggested by the *active blackboard* and *high-resolution buffer* hypotheses?

- What are the neural correlates of explicit visual perception (or visual awareness) and are these guided by feedback effects (for example when focusing on the high-resolution details of an object) as suggested by the Reverse Hierarchy Theory?

- What effects does the inactivation of the different higher visual areas (V4, IT, MT) have on lower level representations? How do these compare under conditions of simple artificial stimuli (e.g. gratings), natural stimuli, and highly cluttered/illusory/occluded stimuli? How do these correlate to subjective visual perception and the performance of vision-related tasks?

2.3 Illusory and occluded contours

Despite living in a cluttered world where the majority of objects we see are partially occluded, we do not have the impression of constantly being surrounded by object fragments. Our visual system appears to have developed the appropriate filling-in or completion mechanisms that crucially allow us to perceive complete objects and make sense of the world. These mechanisms, which compensate for missing or ambiguous information in the retinal image, can be divided into two categories: modal and amodal completion.

Modal completion, the induced perception of contours and surfaces in empty regions, results in perceptually salient effects, such as illusory contours. In contrast, amodal completion, the continuation of contours and surfaces behind occluders, has no visually salient manifestation. Although neither of them have a physical counterpart in the retina, they both show clear neural correlates at different levels of the visual system. Both of these phenomena are closely related to other conspicuous aspects of visual processing, such as feature binding and perceptual grouping. This section provides an overview of the existing experimental evidence for both illusory and occluded contours. The different theoretical approaches and computational models are discussed in the subsequent section.

2.3.1 Experimental evidence

2.3.1.1 Illusory contours

The retinotopic activation of V1 and V2 neurons in response to illusory contours, such as Kanizsa figures, has been reported in fMRI (Maertens et al. 2008), EEG (Seghier and Vuilleumier 2006), MEG (Halgren et al. 2003) and single-cell recording (Lee 2003, Lee and Nguyen 2001) studies. The illusory contour response is weaker, significantly delayed, and only arises in a fraction of V1/V2 cells, in relation to that of real contours. Previous controversy as to whether V1 represents illusory contours seems to have been clarified by the results reported in the previously cited articles. Nonetheless, V1 tends to show a weaker response than V2 and sometimes requires task-related attention to emerge (Lee 2003). Ramsden et al. (2001) also reported orientation reversal between V2 and V1, such that the illusory contour orientation is de-emphasized in V1, while the orthogonal orientation is enhanced. This was suggested to constitute a cortical balancing process which could play an important role in illusory contour signaling, and was later supported by psychophysical data (Dillenburger 2005).

The fact that the illusory contour response does not arise from ordinary feedforward pathways, i.e. retina and LGN, and that it is delayed relative to real contours, suggests the involvement of lateral and feedback connections. Interestingly, the response in V1 emerges later than in V2 (Lee and Nguyen 2001, Halgren et al. 2003, Ramsden et al. 2001, Maertens et al. 2008, Dillenburger 2005) suggesting contour completion in V1 might arise as a consequence of feedback connections from V2. The question arises as to why is it necessary to feed back information to V1 if the illusory contour is already represented in V2. The most likely reason is that V1 neurons' smaller receptive field size provide higher spatial resolution to accurately represent the illusory contour. Bigger receptive field sizes in V2 allow the system to integrate global contextual information which is then fed back to V1 circuits. When required by environmental demands, these circuits can then construct a more precise representation, which explains why illusory contours in V1 sometimes emerge as a consequence of task-related attention (Lee and Nguyen 2001).

Furthermore, a large number of studies have reported neural correlates of illusory contours in

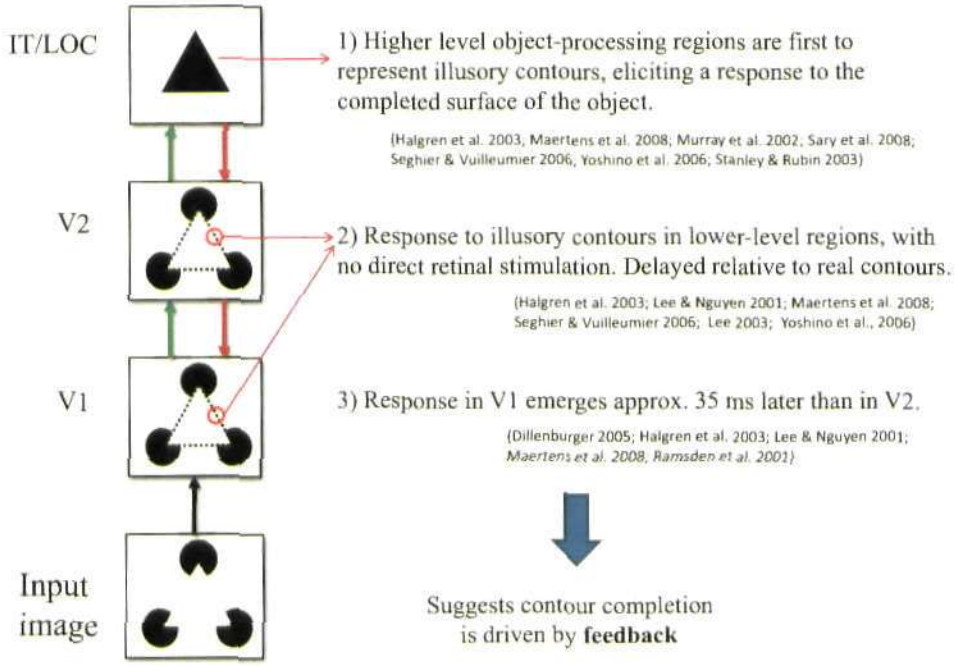


Figure 2.11: Neural correlate and timing of illusory contour emergence along the visual system, with relevant references. Higher-level object processing regions are the first to represent illusory contours after rapid object categorization. Retinotopic responses to illusory contours are then observed in lower level regions such as V1 and V2, which receive no direct retinal stimulation. The response in V1 emerges later than in V2. Overall, the evolution of illusory contour emergence suggests it is driven by feedback connections from higher level regions.

higher level object processing regions such as monkey IT (Sary et al. 2008) and the human equivalent, the LOC region (Halgren et al. 2003, Maertens et al. 2008, Murray et al. 2002, Seghier and Vuilleumier 2006, Yoshino et al. 2006, Stanley and Rubin 2003). A subset of these point out that high-level activity generated by objects containing illusory contours (such as the Kanizsa triangle) is notably similar to the activity of complete objects, despite presenting longer latencies (Stanley and Rubin 2003, Maertens et al. 2008, Sary et al. 2008). IT cells with specific selectivity for illusory contour figures have been reported (Sary et al. 2008). It has been suggested that the large non-retinotopic receptive fields in IT/LOC, which receive bilateral stimulation from massive regions of the visual field, present the best substrate to perform grouping across large distances, such as that required for illusory contours.

A number of key findings strongly substantiate the hypothesis that feedback from high-level extrastriate areas is responsible for subjective contours emerging in lower levels (see Figure 2.11). To begin with, Huxlin et al. (2000) found that monkeys with lesions of IT lost their ability to detect illusory contours. The temporal sequence of events is also consistent with this hypothesis, as LOC/IT regions are the first to signal the appearance of illusory contours, reporting extremely fast response times, such as 90-100 ms (Murray et al. 2002) or 140 ms (Halgren et al. 2003). These studies also show how the visual responses later spread to lower regions including V3, V2 and V1. This is consistent with the multiple processing stages reported by various groups (Yoshino et al. 2006), which distinguish between initial region-based segmentation and later boundary completion processes. In consonance with this finding, an fMRI study (Stanley and Rubin 2003) showed that a Kanizsa figure with well-defined sharp contours and one with blurred contours were represented equivalently in the LOC region. Psychophysical testing demonstrated subjects were indeed able to perceive the sharp and well localized edges in the first case but not in the second, suggesting some other region must be responsible for neurally coding this information, most likely V1 and V2.

The above body of evidence is consistent with the high-resolution buffer hypothesis (Lee 2003), the active blackboard concept (Bullier 2001) and the Reverse Hierarchy Theory (Hochstein and Ahissar 2002) described in section 2.2.2. These approaches hypothesize that V1 might be involved in more complex computations usually attributed to higher-level regions, by interacting with these regions through feedback connections.

2.3.1.2 Occluded contours

It has been suggested amodal completion processes are carried out by the same cortical circuits as illusory contour completion. In this section we will review experimental evidence that indeed highlights the striking similarities between them. These similarities suggest the same high-level feedback mechanisms are being used. In Section 2.3.2 we will further substantiate this argument from a more theoretical point of view.

Neural correlates of occluded contours in early visual areas such as V1 and V2 have been found using fMRI (Weigelt et al. 2007, Rauschenberger et al. 2006), EEG (Johnson and Olshausen

2005) and single-cell recording (Lee and Nguyen 2001, Lee 2003). Rauschenberger et al. (2006) showed how the representation in early visual cortex of an occluded disc evolved from that of a notched disc to one corresponding to a complete disc after approximately 250 ms. The amodal completion process shows temporal properties similar to those of illusory contours, i.e. a delay of approximately 100-200 ms with respect to real contours; although the response tends to be significantly lower than for illusory contours (Lee 2003). This is consistent with the weaker, non-visually salient, perceptual experience associated with occluded contours.

The representation of occluded objects in high-level object recognition areas, such as IT and LOC, has also been repeatedly documented (Hegde et al. 2008, Murray et al. 2006, Weigelt et al. 2007, Hulme et al. 2007). Consistent with this observation, abundant evidence sustains the multistage model of object processing, and shows the temporal representation of occluded contours occurs in a top-down fashion (Murray et al. 2006, Rauschenberger et al. 2006, Weigelt et al. 2007). This is indicative of high-level feedback being responsible for amodal completion in lower regions. However, two diverging interpretations exist as to what exactly is represented by higher-level neurons.

The first interpretation rests upon evidence showing that high-level representations of occlusion are invariant, and have similar time courses and magnitude to those of complete objects (Weigelt et al. 2007, Hulme et al. 2007, Rauschenberger et al. 2006). It therefore suggests that, although the occluded-object and incomplete-object interpretations are both kept alive in lower visual areas, in the higher levels only the occluded-object interpretation persists. This means the high-level neurons represent just the completed object, which becomes the explicit percept. This is consistent with the literature on bistable stimuli which indicates only the conscious percept is represented in high-levels (Fang et al. 2008).

Contrastingly, an fMRI study (Hegde et al. 2008) reported regions in the LOC area which show significantly stronger responses to occluded objects than to unoccluded objects. Along the same lines, Murray et al. (2006) identified within the LOC region a specific object recognition stage which included boundary completion processes. This would suggest the incomplete object is also represented at some stage in this high-level region. The study pointed out that this does not

exclude the involvement of early visual areas in the same contour completion process.

2.3.2 Theoretical and Computational Models

2.3.2.1 Identity hypothesis

Prior to describing the different theoretical approaches to contour completion, it is important to clarify the relation between illusory and occluded contours. Typically, illusory contours are treated as a perceptual phenomenon, because they produce a clear sensorial experience. On the other hand, occluded contours are usually categorized as cognitive phenomena, as they cannot be directly seen, and are better described as being known or inferred.

However, both experimental evidence and theoretical approaches indicate that in fact the same interpolation process is responsible for both seemingly different effects. This controversial claim is known as the identity hypothesis (Kellman 2003). From a representational perspective there shouldn't be any significant difference between a contour which is behind another surface, and a contour which is in front. They cannot be divided into real, perceived and inferred contours, as they all try to represent the reality of the outside world as accurately as possible. The occluded contour is not any more or less real than the illusory contour.

The question then arises as to why such phenomenological differences exist between illusory and occluded contours, if they are both a consequence of the same representational process. This may be due to the fact that different aspects of a scene need to be neurally coded in different ways. Whether a contour is in front of or behind another surface shouldn't affect the process of completing that object to make sense of the world. However, when coding the graspability of a given surface, it is vital to clearly signal occluded non-reachable surfaces, and that may be the role of the modal/amodal phenomenology. On these grounds, and for the purpose of this section, we will treat illusory and occluded contours as stemming from the same contour completion process.

2.3.2.2 Good continuation, relatability and the bipole

One of the Gestalt principles formulated in the early 20th century was the so-called good continuation principle. It describes the innate tendency to perceive lines as continuing in their

established directions, and is considered one of the milestones of perceptual organization. A similar concept, known as relatability, has been suggested as the guiding principle for contour completion in the visual system. It is based on smoothness principles such as contours being differentiable at least once, monotonic, and having bending angles below 90° (Kellman 2003).

An important tool to explain contour completion is the bipole (Neumann and Mingolla 2001, Grossberg et al. 1997, Roelfsema 2006). Although it can have diverging definitions depending on the author, most generally its function is to evaluate the effect a contour element at a given location has on the likelihood of perceiving a contour at a second location. The bipole is typically represented by a characteristic figure-eight shape which describes the coupling strength between the centre unit and surrounding units according to their relative position and orientation. A neural implementation of this concept gives rise to bipole cells, defined as nonlinear grouping operators which receive input from real edges falling inside the bipole lobes (see Figure 2.12). It is strongly grounded on anatomical, psychophysical and physiological data (see Neumann and Mingolla (2001) for a review), and provides a biologically grounded method of implementing good continuation and relatability principles.

This section discusses existing approaches and models that show how these geometric concepts can be implemented with neural mechanisms and how they relate to the physiology and anatomy of the visual cortex.

2.3.2.3 Classification of theoretical models

Two broad theoretical categories for contour completion models have been considered. The first one relies on feedforward processing, and is also known as base grouping. The second deals with recursive models, placing the focus on lateral and feedback connections, and is also known as incremental grouping (Roelfsema 2006, Neumann and Mingolla 2001).

An alternative, though compatible, classification focuses on the specific mechanisms involved in contour completion, and proposes three broad classes: 1) contour interpolation, which inwardly extends two aligned line segments; 2) contour extrapolation, which outwardly extends a segment of a single line segment; and 3) figural feedback, whereby a high-level representation feeds back to complete missing contours (Hafko et al. 2008). The first two classes can

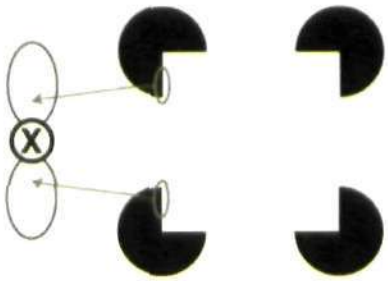


Figure 2.12: The bipole applied to illusory contour formation. The bipole is used to evaluate the effect of a contour element at a given location on the likelihood of perceiving a contour at a second location. It is typically represented by a characteristic figure-eight shape which describes the coupling strength between the centre unit and surrounding units according to their relative position and orientation. A neural implementation of this concept gives rise to bipole cells, defined as nonlinear grouping operators which receive input from real edges (e.g. contours of the pacmen) falling inside the bipole lobes.

sometimes be considered equivalent, and are usually believed to occur as a consequence of feedforward or lateral processing, while feedback connections are the obvious candidate for the third class. As will be discussed further on, evidence suggests it is actually a combination of all three mechanisms that is responsible for contour completion. Note that to avoid confusion we use the term *contour completion* to refer to the general process of filling in missing contours, while we reserve *contour interpolation* for the specific mechanism described above, responsible for contour completion.

2.3.2.4 Feedforward models

This category refers to models based on the feedforward hierarchical architecture of neurons with gradually increasing receptive fields sizes and a spatial overlap between them. Higher processing stages, such as V2, receive converging input from partially activated V1 patches leading to the activation of units with bigger receptive fields which span the gap. These integration units are sometimes referred to as bipole cells, as they group the input from units within the bipole figure-eight geometry, as shown in Figure 2.12. The cooperation stage in Grossberg's model (Grossberg et al. 1997) implements the bipole, although as described further on in this section, it also combines aspects of horizontal processing. Another example of this type of feedforward architecture is HMAX, which was described in Section 2.1.2, although it does not explicitly

implement bipole units.

A similar model which also falls into the base-grouping category focuses on the feed-forward integration of end-stopped responses (Heitger et al. 1998). This approach extends the classical simple and complex cell model to include end-stopped neurons, cells which decrease their response when the stimulus size exceeds the classical receptive field. The response of the oriented end-stopped cells is important as it is associated with figure-ground segregation. Evidence shows that partial occlusion typically generates abrupt terminations at the side of the occluded surface, which can be accounted for by figure-ground segregation mechanisms. Furthermore, the model end-stopped neurons also distinguish and encode the direction of contrast between figure and ground surfaces. It was found that neurons in V2 that respond to illusory contours are sensitive to the direction of contrast, and that this usually matches the occlusion direction. Thus, the model performs contour interpolation of subjective contours by integrating over the end-stopped cells according to the bipole weighting function, and taking into account the contrast-direction selectivity of these cells.

This feedforward processing stage is sometimes referred to as the preattentive phase, and although some highlight the exclusive involvement of feedforward as an advantage, the general consensus is that feedforward processing by itself is insufficient to perform contour completion, *specially across large areas of the visual field. For example, the size of receptive fields in lower* level regions, where contour completion effects are observed, is insufficient to cover the visual field distance ($> 10°$) between the inducer line segments of large illusory figures (Angelucci and Bullier 2003, Sterzer et al. 2006). However, as pointed out at the end of this section, preattentive processing can also be understood as an important initial stage in the more global completion process, which also involves recurrent processing.

Horizontal and feedback models A second category of models, also denoted as incremental grouping models (Roelfsema 2006), take into account the context of the elements involved by making use of horizontal and feedback connectivity. Under this network scheme, neurons stimulated directly from luminance-defined contours provide facilitative interactions to neurons which do not receive direct retinal stimulation. It is usually considered an attentive, and thus

more time-consuming process, which gradually strengthens the responses of features which are perceptually grouped together.

An important concept for these models is the local association field which formalizes the Gestalt good continuation principle. It states that contour elements which are well aligned will tend to be grouped together, by mutually exciting and increasing each other's saliency, while non-collinear elements tend to inhibit each other. Several studies have shown the neural interactions that represent contour elements in V1 and V2 are dictated by the selectivity of horizontal connections which follow the local association field principle (Roelfsema 2006). The local association field provides an alternative method of implementing the bipole principle, based on lateral recurrent circuits (Li 2001) instead of strictly feedforward connections.

Grossberg and colleagues proposed the Boundary Contour System (BCS), one of the most prominent models based on the above principles. The BCS is encompassed by the more general Adaptive Resonance Theory (see Section 2.2.3), and comprises a number of stages that perform detection, competition and cooperation of boundary segments. Later versions of the model combine both feedback, lateral competition and feedforward integration of responses through the so-called bipole cells to achieve contour completion (Grossberg et al. 2007). Here we focus on previous models (Grossberg et al. 1997, Raizada and Grossberg 2001) that show how contour completion is achieved by implementing the bipole property using strictly horizontal connections.

Bipole cells are activated only when both sides of their receptive field are sufficiently stimulated as shown in the schematic diagram of Figure 2.13. Ovals represent pyramidal cells located in layer 2/3 with collinear and co-oriented receptive fields. They are connected to each other via excitatory long-range horizontal synapses. These connections also excite a pool of inhibitory interneurons (black circles) connected via short range synapses to the pyramidal cell. The balance of excitation and inhibition accomplishes the desired bipole property.

When only one of a pair of pacmen is present, the excitation from the inducing pyramidal cell to the target pyramidal cell is not enough to elicit the cell's response. This is because the excitation also targets the inhibitory neurons that balance out the excitation. On the other hand,

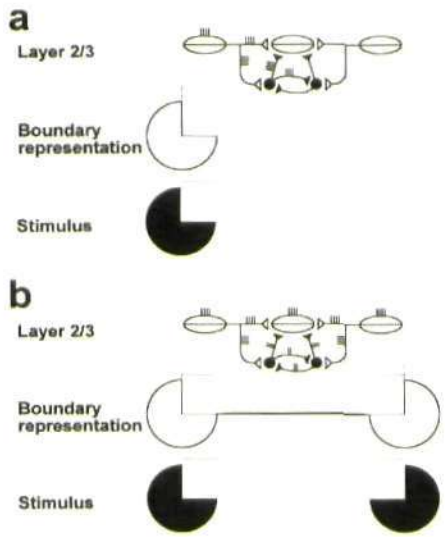


Figure 2.13: Bipole implementation by Raizada and Grossberg (2001). Ovals represent pyramidal cells located in layer 2/3 with collinear and co-oriented receptive fields. They are connected to each other via excitatory long-range horizontal synapses. These connections also excite a pool of inhibitory interneurons (black circles) connected via short range synapses to the pyramidal cell. a) Input from just one side is insufficient to elicit a response in the target pyramidal cell. This is the result of excitation also targeting the inhibitory neurons which balance out the excitation. b) When input arrives from collinearly aligned inducers on either side, the bipole property arises due to the circuits' excitatory/inhibitory balance, leading to contour completion. The target neuron summates inducing excitation arising from neurons at both sides. Additionally, this excitation falls onto the shared pool of inhibitory interneurons, which also inhibit each other, normalizing the total amount of inhibition sent to the target pyramidal neuron.

when inducing excitation comes from neurons at both sides of the target neuron, it summates. Additionally, this excitation falls onto the shared pool of inhibitory interneurons, which also inhibit each other, normalizing the total amount of inhibition sent to the target pyramidal neuron. The combination of summating excitation and normalized inhibition leads to neurons without direct retinal stimulation, representing contours as a result of lateral connectivity.

The key mechanism to achieve contour completion in Grossberg's model is interpolation as defined by the bipole property. Extrapolation is also present in a sense but only when supported by the interpolation bipole mechanism. The model also supports the involvement of figural feedback but is limited to enhancing contours already formed by lateral connections, in what has been called the attentive stage (Halko et al. 2008). As explained in Section 2.1.2, during the attentive stage feedback can only provide subthreshold modulatory effects but never directly drive the cell. This is consistent with the idea that feedback connections from areas V2, V4 and IT have a role in shaping the local association field (Roelfsema 2006).

A similar theoretical approach, which places a stronger emphasis on figural feedback, was proposed by Lee (2003). It suggested illusory contours originated in higher levels with bigger receptive fields which could take into account a greater contextual range. This spatially diffuse activity can act as a top-down prior which feeds back to V1's high resolution buffer. The local association field implemented via V1 horizontal connections can then refine the feedback to construct spatially sharp and precise contours. Figure 2.14 summarizes the described contour completion process resulting from the interaction between feedback and lateral connections.

Further evidence suggests that, although global feedback can interact with local boundary completion, these two processes are distinguishable and independent. Feedback, which emerges after processing the available partial information during feedforward recognition, leads to imprecise boundary completion unless guided by the appropriate local cues dictated by contour relatability (Kellman et al. 2003). This was shown using the dot localization paradigm, whereby an occluded image is presented, followed by a dot in front of the occluder which is rapidly masked. Subjects need to judge whether the dot is perceived to appear inside or outside of the occluder's contour.

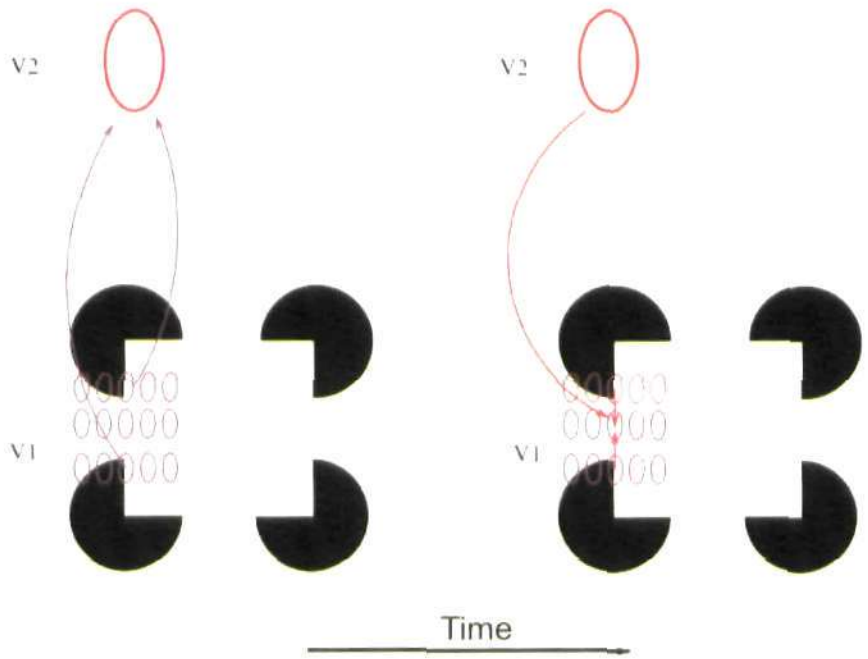


Figure 2.14: Contour completion resulting from the interaction between global feedback and local horizontal connections as proposed by Lee (2003). Left: A V2 neuron receives feedforward connections from a set of V1 neurons of a particular orientation activated by real edges. Right: The V2 neuron projects feedback to the same set of V1 neurons. The excited V2 neuron broadcasts the possibility that an extended contour exists to all V1 neurons. This distributed feedback signal introduces the global context that motivates the completion of contour by the V1 neurons based on local evidence precisely represented in V1.

Results showed the representation of boundaries was highly precise and accurate only when the contour could be predicted by local relatability cues. However, when completion had to be predicted from long-range contextual information, such as from global symmetry patterns, the precision strongly diminished. This suggests feedback alone can provide widespread activity indicating the presence of subjective contours, but is insufficient to perform accurate contour completion.

Integration of multiple mechanisms Recent reviews of illusory contour formation suggest that it results from the interaction between the different proposed mechanisms, i.e. extrapolation, interpolation and figural feedback. Evidence suggests they all play a role in subjective contour perception, although the significance of their contribution may vary according to the conditions of the stimuli. This addresses previous conflict between evidence in support of bottom-up versus top-down processing. Feedforward and horizontal connections would be involved in interpolation and extrapolation processes, which interact with the high-level figural feedback, as supported by psychophysical, physiological and anatomical data. For a detailed review see Halko et al. (2008).

This interaction of mechanisms is consistent with models where the input image is preattentively segmented based on Gestalt principles and subsequently processed following high-level focal attention (Grossberg and Raizada 2000, Marr 1982). Under this perspective the initial feedforward base grouping would generate the high-level percept which would then attentionally guide, from coarse to fine scale, the local incremental grouping process that leads to illusory contour formation. This could in turn provide more detailed representation, which could improve higher level object recognition (Hochstein and Ahissar 2002). For example, the fact that perceptual grouping does not occur during the inattentional blindness condition (inability to perceive something that is within one's direct perceptual field as a result of a lack of attention) of an image provides further support for this conclusion (Roelfsema 2006). Overall, the different approaches described in this section can be integrated into a common global recurrent process spanning several regions of the visual system, each of which interacts in parallel to achieve the completion effect.

To conclude, it is important to stress the significance of subjective contours not only for perceptual purposes, but also for related functions such as action and other cognitive processes. Contour completion incorporates missing information which leads to a more unfaithful representation of the two-dimensional input image, but a more accurate and reliable representation of the surrounding physical environment. From this perspective, they are not merely illusions which should be discarded, but are in fact representations which bring our perceptual experience closer to reality (Kellman 2003).

2.3.2.5 Key questions

Overall, the contour completion phenomenon poses an intriguing and exciting challenge to the scientific community, with many open questions still to be resolved. Answering these questions about what appears to be one of the key elements in visual perception will undoubtedly constitute an enormous contribution to our understanding of this and other related fields.

- Are modal (illusory) and amodal (occluded) completion effects mediated by common neural mechanisms (identity hypothesis)? If so, how are the striking phenomenological differences represented in cortex?

- Why do studies show contradictory evidence in relation to whether subjective contours are represented in V1 or not? Are these inconsistencies a consequence of task/behaviour-related demands (e.g. more visual precision is required for certain tasks) or internal methodological differences?

- What are the neural mechanisms that mediate the integration of bottom-up, horizontal and feedback information in order to generate subjective contours? Does the reliability/unambiguity of each of the sources determine the weight of its contribution? If so, how is the weighting process neurally coded?

- If feedback from high-level object-related areas is responsible for lower level contour completion effects, does this feedback proceed in a serial sequence (e.g. IT-V4-V2-V1) or via parallel streams (e.g. IT-V1, V4-V1, V2-V1)? In the latter case, a) how are the

different feedback sources integrated? b) if one of the sources is inactivated (e.g. IT) are the remaining regions sufficient to generate the subjective contour in V1?

- Are illusory and occluded objects represented in high-level regions in a different way to complete objects? If so, are they represented with their missing parts, or as complete objects but with a watermark indicating they are occluded/illusory?

- For Kanizsa figures with ambiguously defined inducers, e.g. rounded corners, the illusory contour is no longer perceived (based on psychophysical data). Is feedback equivalent in Kanizsa figures with ambiguously vs. precisely defined inducers? Is the perceptual difference due to the lack of local precise cues which prevents horizontal connections from forming the illusory contour? If the rounded corners were gradually transformed into straight corners, a) at what point would the illusory contour be perceived?, and b) how would this correlate to the neural representation in V2 and V1?

- What neural processes are being activated when a human observer decides to voluntarily change the perception of a Kanizsa figure to that of individual pacmen? Is feedback responsible for inhibiting the generation of the illusory contours?

2.4 Original contributions in this chapter

- Review evidence and identify key questions on the role of cortical high-level feedback in object perception.

- Analyze and compare the different functional interpretations of feedback and identify points of convergence between them.

- Review evidence and identify key questions on the representation in the visual system of illusory and occluded contours.

Chapter 3

Bayesian networks and belief propagation

As described in Chapter 2, the classical feedforward processing model fails to capture many observed neurophysiological phenomena, and thus is gradually being replaced by a more global and integrative approach which relies on feedback connections. However, theoretical and computational models still strive to accommodate feedback connections and the different observed contextual effects within a single general theoretical framework. The probabilistic inference approach described in this chapter attempts to solve this problem. Results presented in this thesis are based on this methodological approach, and more specifically on belief propagation in Bayesian networks. Thus, Section 3.1 offers an introduction to Generative models and Bayesian inference, providing the theoretical background and roots of this approach. Section 3.2 reviews evidence that supports this framework as being a good candidate for modelling the visual cortex. Section 3.3 defines and formulates mathematically both Bayesian networks and the belief propagation algorithm, and includes an illustrative example. Finally, existing theoretical and computational models based on belief propagation are described in Section 3.4.

3.1 The Bayesian brain hypothesis

3.1.1 Generative models

It has long been appreciated that information falling on the retina cannot be mapped unambiguously back onto the real-world; very different objects can give rise to similar retinal stimulation, and the same object can give rise to very different retinal images. So how can the brain perceive and understand the outside visual world based on these ambiguous two-dimensional retinal images? A possible explanation comes from the generative modelling approach, which has as its goal the mapping of external causes to sensory inputs. By building internal models of the

world the brain can generate predictions and explain observed inputs in terms of inferred causes. Formulating perception as a process based on generative models which employs Bayesian probability theory to perform inference is known as the *Bayesian brain hypothesis* (Friston 2010).

From this perspective the brain acts as an inference machine that actively predicts and explains its sensations. The basic idea is that making predictions is an effective strategy for discovering *what's out there*, and for refining and verifying the accuracy of representations of the world; in this way the world can act as its own check. Mismatches between expected and actual sensory experience allow us to identify the things that we don't know about, and hence fail to predict. This information can then be used in the creation and refinement or updating of internal representations or models of the world, which in turn lead to better predictions.

A natural consequence of these ideas is that the processing architecture and sensitivities should reflect the structure and statistics of natural sensory inputs. This suggests the visual cortex might have evolved to reflect the hierarchical causal structure of the environment which generates the sensory data (Friston and Kiebel 2009, Friston 2005, Friston et al. 2006, Friston 2010) and that it can consequently employ processing analogous to hierarchical Bayesian inference to obtain the causes of its sensations, as depicted in Figure 3.1.

3.1.2 Bayesian inference

Making inferences about causes depends on a probabilistic representation of the different values the cause can take, i.e. a probability distribution over the causes. This suggests replacing the classical deterministic view, where patterns are treated as encoding features (e.g. the orientation of a contour), with a probabilistic approach where population activity patterns represent uncertainty about stimuli (e.g. the probability distribution over possible contour orientations). The Bayesian formulation provides the tools to combine probabilistic information, i.e. prior knowledge and sensory data, to make inferences about the world.

According to the Bayesian formulation the generative model is decomposed into two terms: the likelihood function or the probability that certain causes would generate the sensory input in question; and the prior or unconditioned marginal probability of those causes. The likelihood model, which maps causes to sensations, can be inverted using the Bayes theorem, yielding the

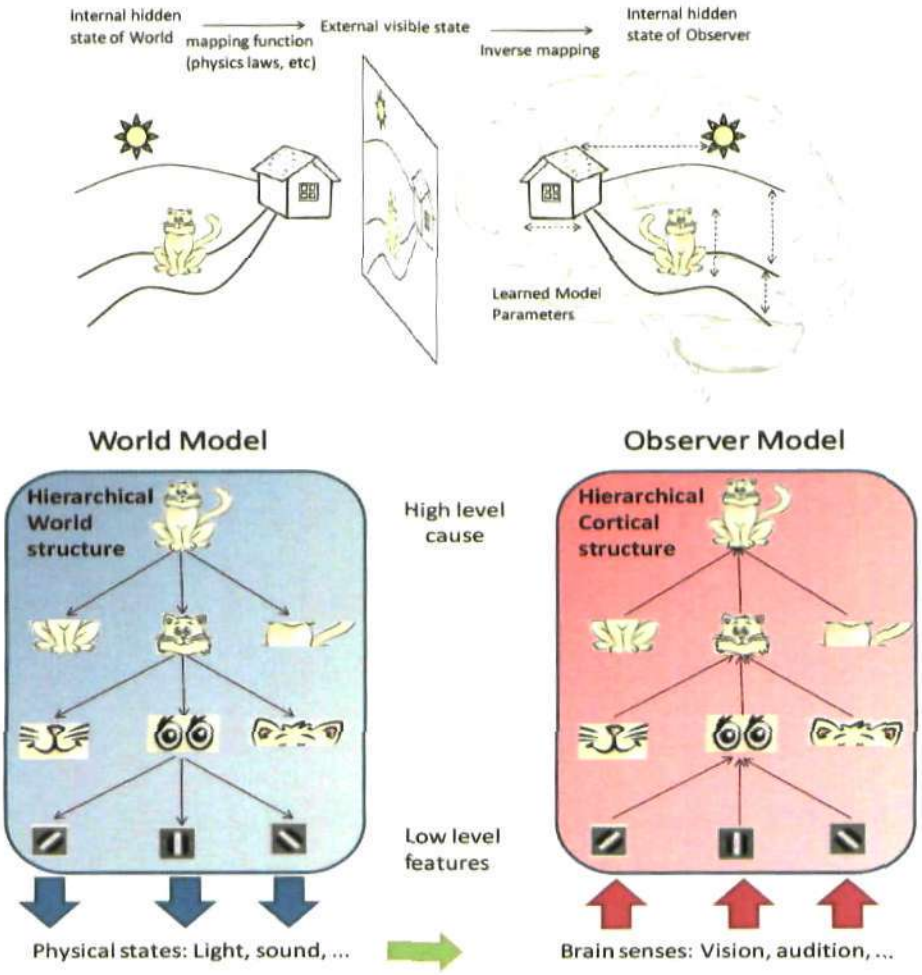


Figure 3.1: Learned internal model in visual cortex reflects hierarchical causal structure of the environment which generates the sensory input. The ambiguous information provided by sensory inputs (e.g. 2D retinal image) is only a function of the internal state of the world (e.g. 3D objects). The brain (observer) needs to inversely map this function as precisely as possible to generate an accurate internal representation of the world. The hierarchical organization of the brain suggests it has evolved to reflect the inherent hierarchical structure of the world.

posterior probability of the causes given the inputs (mapping from sensations to causes). This can be written as:

$$P(C|I) = \frac{P(I|C) \cdot P(C)}{P(I)} \tag{3.1}$$

where $P(C|I)$ represents the posterior probability of the causes C given the input I, for example the probability over the different physical causes given a particular retinal image; $P(I|C)$ represents the likelihood of the input I given the causes C, for example the probability of a given retinal image having been generated by one or another of the potential different physical causes; $P(C)$ represents the prior probability of the causes C, for example the different physical states of the world; and $P(I)$ simply represents a normalization factor.

3.1.3 Free-energy principle

The free-energy principle proposed by Friston (Friston and Kiebel 2009, Friston 2005, Friston et al. 2006, Friston 2010) conceptualizes the brain as an adaptive system which tries to resist a natural tendency to disorder, or entropy. Entropy can also be understood as a measure of uncertainty or surprise, thus informally, the system needs to avoid surprises to ensure its state remains within physiological bounds. One of the main characteristics of biological systems is that they maintain their internal states within operational bounds, even with constantly changing environments.

However, how can a system know if its sensations are surprising? The free energy principle provides a framework to do this as the free energy of a system is an upper bound on surprise. Thus by minimizing free energy, the system is implicitly minimizing surprise. Importantly, free energy can be evaluated because it depends on two probability densities which are available to the system: the recognition density and the conditional or posterior density.

The recognition density, $P(\vartheta|\mu)$, provides a probabilistic representation of the causes, ϑ, of a particular stimulus, given a set of internal states, μ. In the brain these internal states hypothetically correspond to neuronal activity and synaptic weights. The conditional density, $P(\tilde{s}, \vartheta|m)$, provides the joint probabilistic representation of causes, ϑ, and sensory signals, $\tilde{s}$. It is based

on a probabilistic generative model, m, which captures the dependencies between causes and sensory data, and can thus generate sensory samples from given causes, and likewise obtain a posterior distribution of causes given the sensory input. The generative model is hypothesized to be implicitly imprinted in the hierarchical structure of the brain.

The theory accommodates several aspects of brain function in terms of optimizing the different parameters in order to minimize the free energy of the system. For example, perception is understood as the process of minimizing free energy with respect to the neuronal activity (encoded as part of the internal state, μ), which entails maximizing the posterior probability of the recognition density. The recognition density therefore becomes an approximation of the true posterior density. This is equivalent to the Bayesian inference approach described previously in this section. Similarly, learning or plasticity in the brain is explained as the optimization of synaptic weights, also encoded by the internal state variable, μ. These two processes minimize free energy by changing the recognition density, which modifies the expectations about sensory data, but without modifying sensory data itself.

On the other hand, action is understood as a process of active inference, aimed at modifying sensory data so that it conforms to the predictions or expectations made by the recognition density. Increasing the accuracy of predictions also reduces the free energy of the system. Broadly speaking, *the prediction error (i.e. sensations minus predictions), and thus free energy, can* be minimized by either changing the sensory input through action, or changing the predictions through perception and learning. For a comprehensive description of the mathematical formulation of free energy minimization the reader is referred to Friston and Kiebel (2009).

The free energy formulation was originally developed to deal with the problem of obtaining exact inferences in complex systems. It tackles the problem by converting it into an easier optimization problem. The inversion of the likelihood function (based on the Bayes theorem) to infer the posterior distribution over causes, thus becomes an optimization problem which consists of minimizing the difference between the recognition and the posterior densities to suppress free energy. This technique can be described as a type of variational Bayesian method (Beal 2003, Friston and Kiebel 2009, Winn and Bishop 2005), also called ensemble learning.

These methods provide an analytical approximation to the posterior probability of intractable Bayesian inference problems.

In summary, the free energy principle provides a unifying framework for the Bayesian brain and predictive coding approaches, which understand the brain as an inference machine trying to optimize the probabilistic representation of what caused its sensory input. As stated by Friston (2010), the theory can be implemented by many different schemes, most of which involve some form of hierarchical message passing or belief propagation among regions of the brain.

The model proposed in this thesis describes such a hierarchical message passing scheme, and thus is theoretically grounded on the free energy principle and the Bayesian brain hypothesis. Particularly, the focus of this thesis is on Bayesian networks, a type of graphical model which represents the causal dependencies present in generative models; and the Bayesian belief propagation algorithm, which performs inference in this type of network. A more formal definition and the relevant mathematical formulation of Bayesian networks and belief propagation is included in Section 3.3.

3.1.4 Origins

One of the first people to propose formulating perception in terms of a generative model was Mumford, who based his ideas on Grenader's pattern theory and earlier suggestions by Helmholtz (Mumford 1996). Applied to visual perception, this theory states that what we perceive is not the true sensory signal, but a rational reconstruction of what the signal should be. The ambiguities present in the early stages of processing an image never become conscious because the visual system finds an explanation for every peculiarity of the image. Pattern theory is based on the idea that pattern analysis requires pattern synthesis, thereby adding to the previous purely bottom-up or feedforward structure a top-down or feedback process in which the signal or pattern is reconstructed.

The Helmholtz machine (Dayan et al. 1995) extended these ideas by implementing inferential priors using feedback. Here, the generative and recognition models were both implemented as structured networks whose parameters had to be learned. The connectivity of the system is based on the hierarchical top-down and bottom-up connections in the cortex. This layered

hierarchical connectionist network provides a tractable implementation to computing the exponential number of possible causes underlying each pattern, unlike other approaches such as the Expectation-Maximization algorithm, which runs into prohibitive computational costs. The key insight is to rely on using an explicit recognition model with its own parameters instead of using the generative model parameters to perform recognition in an iterative process.

In recent years, the Bayesian brain hypothesis has become increasingly popular, and several authors (Friston 2005, Dean 2006, Lee and Mumford 2003, Rao 2006, Deneve 2005, Litvak and Ullman 2009, Steimer et al. 2009, Hinton et al. 2006) have elaborated and extended this theory. Many of their contributions are described in this chapter. One of the main reasons for the rising recognition of the Bayesian brain hypothesis is its ability to accommodate disparate experimental results and existing models within a common framework, as will be illustrated in the following sections.

3.2 Evidence from the brain

The Bayesian brain model maps well onto anatomical, physiological and psychophysical aspects of the brain. Visual cortices are organized hierarchically (Felleman and Van Essen 1991) in recurrent architectures using distinct forward and backward connections with functional asymmetries. While feedforward connections are mainly driving, feedback connections are mostly modulatory in their effects (Angelucci and Bullier 2003, Hupe et al. 2001). Evidence shows that feedback originating in higher level areas such as V4, IT or MT, with bigger and more complex receptive fields, can modify and shape V1 responses, accounting for contextual or extra-classical receptive field effects (Guo et al. 2007, Harrison et al. 2007, Huang et al. 2007, Sillito et al. 2006). Chapter 2 describes these aspects in more detail. As we will see in this section, hierarchical generative models are reminiscent of the described cortical architecture, sharing many structural and connectivity properties.

In terms of the neural mechanisms involved, although it is not yet practical to test the proposed framework in detail, there are some relevant findings from functional magnetic resonance imaging (fMRI) and electrophysiological recordings. Murray et al. (2004) showed that when local information is perceptually organized into whole objects, activity in V1 decreases while activ-

ity in higher areas increases. They interpreted this in terms of high-level hypotheses or causes *explaining away* the incoming sensory data. Further, Lee and Mumford (2003) studied the temporal response of early visual areas to different visual illusions, concluding that there are increasing levels of complexity in information processing within V1, and that low-level activity is highly interactive with the rest of the visual system. Results of both experiments are consistent with the generative modelling approach.

The generative model is also in agreement with evidence suggesting that the representations activated along the ventral pathway that are activated during mental imagery and visual perception are surprisingly similar (Reddy et al. 2010, Ishai 2010). In fact, Slotnick et al. (2005) showed that visual mental imagery can evoke topographically organized activity in striate an extrastriate cortex, suggesting the involvement of feedback connections from higher-level object-representation regions (Reddy et al. 2010).

The model is also consistent with evidence showing feedback from higher levels acts to reduce responses in lower levels (Alink et al. 2010, Murray et al. 2004, Sterzer et al. 2006, Harrison et al. 2007, Rao and Ballard 1999). This is related to the predictive coding approach (Section 2.2.3), which is a particularization of hierarchical Bayesian inference in generative models. The reduction in response can be explained either by the reduction in feedforward error-detection populations, as a consequence of more accurate high-level predictions, or by a refinement of the belief maintained at the different levels, due to the reduction of activity coding for features inconsistent with high-level predictions. Furthermore, predictive coding models have been shown to be successful in explaining several phenomena observed in cortex, such as repetition suppression (Friston et al. 2006), biphasic responses in LGN (Jehee and Ballard 2009), end-stopping effects (Rao and Ballard 1997, Rao 1999) and a wide variety of V1 extra-classical receptive field effects, including surround suppression and facilitation (Spratling 2010).

The model also accommodates evidence, such as the reduction of V1 activity when higher areas are inactivated (Hupe et al. 2001, Huang et al. 2007, Galuske et al. 2002, Angelucci and Bullier 2003), which suggest that feedback acts to enhance lower level activity consistent with the high-level percept. This is consistent with biased competition and attentional interpretations of

feedback, which can also be accommodated within Bayesian inference theory (Chikkerur et al. 2009, Spratling 2008b, Friston 2010). These results are explained as an increase in the belief or prediction populations, as a consequence of an enhancement of features consistent with the global percept.

Moreover, the Bayesian framework is also compatible with basic synaptic physiology such as Hebbian plasticity, which results from the optimization of the generative model parameters in order to reduce prediction error (Friston et al. 2006). A recent study (Nessler et al. 2009) further showed how a winner-take-all network of spiking neurons implementing a spike-timing-dependent plasticity rule could be understood in terms of a hierarchical generative model which *discovered the causes of its input*.

Research has also made progress in accommodating the probabilistic framework at a neuronal processing level, describing how simple spiking neuron responses and population codes can represent probability distributions and implement inference (Pouget et al. 2003, Zemel et al. 2004, Deneve 2008a,b, Ma et al. 2006, Wu and Amari 2001). A recent outstanding publication (Soltani and Wang 2010) demonstrated how neuronal synaptic computations could underlie probabilistic inference by integrating information from individual cues. The model, validated on data from an experiment on a monkey performing a categorization task, showed how synapses, based on reward-dependent plasticity, naturally encode the posterior probability over different causes given the presentation of specific cues.

Our understanding of the psychophysics of action and perception has also strongly benefited from Bayesian inference approaches. These have provided a unifying framework to model the psychophysics of object perception (Kersten et al. 2004, Knill and Richards 1996, Yuille and Kersten 2006), resolving its complexities and ambiguities by probabilistic integration of prior object knowledge with image features. Interestingly, visual illusions, which are typically interpreted as errors of some imprecise neural mechanism, can in fact be seen as the optimal adaptation of a perceptual system obeying rules of Bayesian inference (Geisler and Kersten 2002). Similarly, Weiss and Adelson (1998) presented a Bayesian model of motion perception which predicted a wide range of psychophysical results, including a set of complex visual illusions, by

combining information from different image regions with a probabilistic prior favouring slow and smooth velocities. In further support of this view, Kording and Wolpert (2004) concluded that the central nervous system also employs a Bayesian inferential approach during sensorimotor learning.

Probabilistic models are currently widely used to successfully capture different aspects of brain function, and provide a unifying perspective across a broad range of domains and levels of abstraction. They are not limited to modelling perception, and have been employed to explain other cognitive functions such as psychological conditioning, semantic memory, and decision-making (Chater et al. 2006). For example, a recent study employs a probabilistic inference computational model, based on the neural representations in prefrontal cortex, to explain decision making during social interactions (Yoshida et al. 2010).

3.3 Definition and mathematical formulation

In this section we define and formulate the mathematical tools used to develop the model in this thesis, namely Bayesian networks and belief propagation. These provide a specific implementation of the theoretical principles described in Section 3.1, i.e. the Bayesian inference and generative model framework. A body of experimental evidence highlighting the similarities between this approach and a set of functional, anatomical, physiological and biological properties of the brain has been presented in Section 3.2.

This section first introduces basic probability theory concepts, and then describes what a Bayesian network is and how belief propagation works, with the aid of a practical example. Subsequent subsections describe two challenging aspects of belief propagation: combining information from multiple parents and dealing with loops in the network using approximate inference methods.

3.3.1 Probability theory

*Before describing Bayesian networks in detail, and to facilitate understanding, this section in*troduces some essential concepts and terminology from probability theory. Note capital letters denote random variables, e.g. X, Y, while lower-case letters denote specific values of a random

variable, e.g. $P(x)$ is equivalent to $P(X = x)$.

Joint probability Given a set of random variables $\bar{X} = \{X_1, ..., X_n\}$, the joint probability distribution $P(X_1, ..., X_n)$ defines the probability of the events specified by the variable states $\bar{x} = (x_1, ..., x_N)$ occurring together (in conjunction), and satisfies the following property,

$$\sum_{x_1, ..., x_n} P(x_1, ..., x_n) = 1 \tag{3.2}$$

Marginal probability Given a joint probability distribution $P(X_1, ..., X_n)$, the marginal probability for a subset of variables $\bar{Y} = \{Y_1, ..., Y_n\} \subset \bar{X}$ is given by

$$P(y_1, ..., y_n) = \sum_{(x_1, ..., x_n) \notin \bar{Y}} P(x_1, ..., x_n) \tag{3.3}$$

The marginal probability for a given variable is therefore

$$P(x_i) = \sum_{(x_1, ..., x_n) \setminus x_i} P(x_1, ..., x_n) \tag{3.4}$$

The marginalization process, also called *variable elimination*, entails summing over all the possible values of the variables we want to eliminate from the resulting marginal probability distribution.

Conditional probability Given two disjunctive sets of variables $\bar{X}$ and $\bar{Y}$, the conditional probability of $\bar{Y}$ given $\bar{X}$ is defined as

$$P(\bar{y}|\bar{x}) = \frac{P(\bar{y}, \bar{x})}{P(\bar{x})} \tag{3.5}$$

Conditional independence Two sets of variables $\bar{X}$ and $\bar{Y}$ are conditionally independent given a third set $\bar{Z}$ if

$$P(\bar{y}|\bar{x}, \bar{z}) = P(\bar{y}|\bar{z}) \tag{3.6}$$

In this case it can be said that set $\bar{Z}$ separates sets $\bar{X}$ and $\bar{Y}$, written as $\bar{X} \perp\!\!\!\perp \bar{Y}|\bar{Z}$.

Factorization From the definition of conditional probability it follows that the joint probability distribution of a set of hierarchically organized variables $\bar{X}$ can be factorized as follows (also referred to as the *chain rule*),

$$P(x_1,...,x_n) = P(x_1|x_2,...,x_n) \cdot P(x_2|x_3...,x_n) \cdot ... \cdot P(x_n) = \prod_i P(x_i|x_{i+1},...,x_n) \tag{3.7}$$

The different conditional probability terms can be simplified according to conditional independence assumptions.

Bayes theorem Given two sets $\bar{X}$ and $\bar{Y}$, the conditional probability of $\bar{Y}$ given $\bar{X}$ (also called the posterior probability) satisfies the following equation,

$$P(\bar{y}|\bar{x}) = \frac{P(\bar{y}) \cdot P(\bar{x}|\bar{y})}{P(\bar{x})} \tag{3.8}$$

where the conditional probability $P(\bar{x}|\bar{y})$ is also called the *likelihood*; the marginal probability $P(\bar{y})$ is also called the *prior*; and the marginal probability $P(\bar{x})$ acts as a *normalization constant*.

The marginalization and factorization of the joint probability distribution, together with the application of the Bayes theorem, are the three key elements of the belief propagation algorithm described in the following section.

3.3.2 Bayesian networks

A Bayesian network is a specific type of graphical model, more specifically a *directed acyclic graph*, where each node in the network represents a random variable, and arrows establish a causal dependency between nodes. Therefore, each arrow represents a conditional probability distribution $P(X|\Pi_X)$ which relates node X with its parents Π_X. Crucially, the network is defined such that the probability of a node X being in a particular state depends only on the state of its parents, Π_X. Consequently, a Bayesian network of N random variables X_i defines a joint probability distribution which can be factorized as follows,

$$P(X_1,...,X_N) = \prod_i P(X_i|\Pi_{X_i}) \tag{3.9}$$

Note for nodes without parents (root nodes), the conditional probability of X_i is equal to its prior probability, i.e. $P(X_i|\Pi_{X_i}) = P(X_i)$. Thus, defining the whole structure of a Bayesian network requires specification of the conditional probability distribution of each node with parents, $P(X_i|\Pi_{X_i})$, plus the prior probability distributions of all root nodes, $P(X_{root})$.

More formally, a Bayesian network is a pair $B = (G,P)$, where

- $G = (V,A)$ is an acyclic directed graph with $V = \{X_1,X_2,...,X_n\}$, a set of nodes (vertices); and $A = \subseteq V \times V$, a set of arcs defined over the nodes;
- $P(V)$, a joint probability distribution over V, given by Equation (3.9).

An explanation of why the graph is denoted as *acyclic* and *directed*, and why these two properties are important, can be found further down in this section after introducing a clarifying example.

3.3.2.1 An illustrative example

Figure 3.2 shows a Bayesian network with six random variables representing a toy model scenario which can be used to illustrate the above concepts. For simplicity we use discrete binary variables, i.e. each variable can be in either of two states, true or false. However, in a real scenario these variables are typically either continuous, or discrete with several states.

The scenario assumes the presence of big waves in the sea is a consequence of two causes: the presence of gales, as strong winds are associated with large wind-generated waves; and whether the moon is aligned with the sun or not. When the moon and the sun are aligned (which occurs during full moon and new moon periods) their gravitational force is combined increasing the amplitude of tidal waves. The presence of big waves is represented by the variable *Waves* (W); the presence of gales is represented by the variable *Gales* (G); and whether the moon is aligned with the sun or not is represented by the variable *Moon* (M).

Because both *Gales* and *Moon* have no parent nodes, they are considered to be root nodes, and

thus require a prior probability distribution. The prior distributions, $P(G)$ and $P(M)$, indicate that, with no other information available, it is more likely that *Gales* are not present (0.8 vs. 0.2); while both states of the *Moon* are equally likely (0.5).

The model assumes when both high-level causes, *Gales* and *Moon*, are present ($G = 1, M = 1$), the probability of *Waves* is higher than when either of the causes is present by itself e.g. *Gales* but no *Moon* ($G = 1, M = 0$). When presented exclusively, *Gales* is considered to have a stronger effect over the generation of *Waves* than *Moon*. All this information is captured by the conditional probability distribution, in this case a conditional probability table (CPT) as variables are discrete, over the states of *Waves* given the states of *Gales* and *Moon*, i.e. $P(W|G,M)$.

At the same time, *Waves* acts as the cause of the two lower level effects: the presence of fishing activity, which is affected negatively by big waves, e.g.fishermen at a pier/beach or small fishing boats; and the presence of surfing activity, a sport which strongly benefits from big waves. The presence of fishing activity is denoted by the variable *Fishing* (F), while the presence of surfing activity is denoted by the variable *Surfing* (S).

Crucially, the state of the parent node, *Waves*, is a determinant factor for the state of both child nodes, *Fishing* and *Surfing*. The causality dependency between the state of the node *Fishing* with respect to the state of its parent node *Waves* is given by the CPT $P(F|W)$. Analogously, $P(S|W)$ represents the conditional probability over the states of the node *Surfing* given the state of the node *Waves*.

Using the more formal definition, the Bayesian network in Figure 3.2 can be described as $B = (G,P)$, where

- $G = (V,A)$ is a directed acyclic graph with a set of vertices $V = \{G,M,W,F,S\}$; and a set of arcs A = {(G,W),(M,W),(W,F),(W,S)};
- P is the joint probability distribution over V given by,

$$P(G,M,W,F,S) = P(S|W) \cdot P(F|W) \cdot P(W|G,M) \cdot P(G) \cdot P(W) \tag{3.10}$$

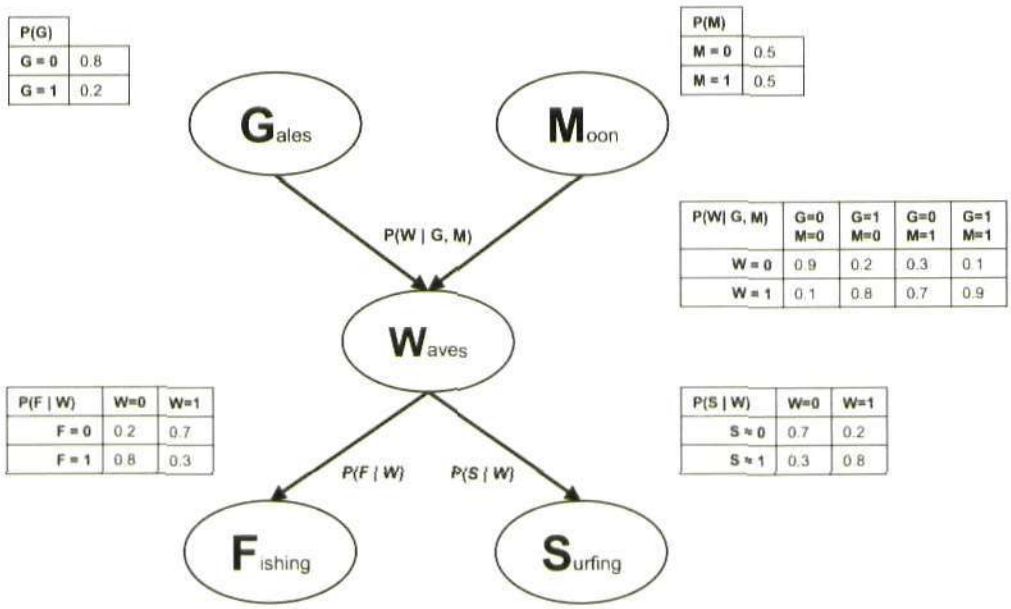


Figure 3.2: Toy model Bayesian network. The scenario assumes the presence of big waves in the sea, *Waves* (W), is a consequence of two causes: the presence of gales, *Gales* (G); and whether the moon is aligned with the sun or not, *Moon* (M). At the same time, *Waves* acts as the cause for the two lower level effects: the presence of fishing activity, *Fishing* (F); and the presence of surfing activity, *Surfing* (S). The prior distributions, $P(G)$ and $P(M)$, indicate that, without any other information available, it is more likely that *Gales* are not present (0.8 vs. 0.2); while both states of the *Moon* are equally likely (0.5). The conditional probability over the states of *Waves* given the states of *Gales* and *Moon* is represented in the conditional probability table $P(W|G,M)$. The causality dependency between the state of the node *Fishing* with respect to the state of its parent node *Waves* is given by the CPT $P(F|W)$. Analogously, $P(S|W)$ represents the conditional probability over the states of the node *Surfing* given the states of the node *Waves*.

Note the example described is just a toy scenario which does not accurately reflect the physical factors affecting wave generation, or the effects waves may have on surfing and fishing activity. Any real life situation is practically impossible to capture using such a reduced number of variables and states. However, the analogy with a real-world situation is useful to explain the different mathematical constructs in this section.

3.3.2.2 Directional separation and explaining away

Graphical models can be divided into two categories: directed and undirected. Undirected graphical models, also called Markov random fields, have a simple definition of independence. Two sets of nodes A and B are conditionally independent given a third set, C, if all paths between the nodes in A and B are separated by a node in C. By contrast, directed graphical models (Bayesian networks), have a more complicated notion of independence, which takes into account the directionality of the arcs. Directionality, however, has several advantages. The most important is that causality is clearly defined, such that an arc from A $\rightarrow$ B indicates that A *causes* B. This facilitates the construction of the graph structure, and the parameter learning process or fitting to data. Not all causal relationships captured with directed graphical models can be represented using undirected graphical models, and vice versa (Pearl 1988, Murphy 2002).

Before describing directional separation in Bayesian networks, it is important to define the concept and the different types of *evidence*. An evidence function that assigns a zero probability to all but one state is often said to provide *hard evidence*; otherwise, it is said to provide soft evidence (e.g. 90% probability of being true and 10% probability of being false). Hard evidence on a variable X is also often referred to as instantiation of X or to X being observed or known. Note that, as soft evidence is a more general kind of evidence, hard evidence can be considered a special type of soft evidence. If the distinction is unimportant we will leave out the *hard* or *soft* qualifier, and simply talk about evidence.

Due to the directionality of arcs, there are three different types of connections in Bayesian networks:

- *Serial connections*: For example, the connection $M \rightarrow W \rightarrow S$ in Figure 3.2. Node S is conditionally dependent on node W, and node W is conditionally dependent on node M. This means entering evidence at nodes M or S will update the probability distribution of node W. However, entering hard evidence in node W blocks or directionally separates (*d-separates*) nodes M and S.

  In other words, nodes M and S are conditionally independent given hard evidence in node W, also written as $M \perp\!\!\!\perp S \mid W$. Thus, information may flow through serial connections unless the state of the middle variable (W) is known. Intuitively speaking, given that we already know the size of the waves (W), the moon alignment (M) does not affect the presence of surfing activity (S).

- *Diverging connections*: For example, the connection $F \leftarrow W \rightarrow S$ in Figure 3.2. Child nodes F and S are conditionally dependent on parent node W, thus entering evidence on W will modify the probability distribution in nodes F and S. However, knowing the state of W blocks (*d-separates*) nodes F and S.

  Therefore, nodes F and S are conditionally independent given hard evidence at node W, i.e. $F \perp\!\!\!\perp S \mid W$. Hence, information may flow along diverging connections, unless the state of the middle node W is known. Intuitively, if we don't know the state of the waves (W), the presence of fishing activity (F) could provide us some information about the presence of surfing activity (S). However, once we know the exact state of the waves (W), the presence of fishing activity (F) does not affect the presence of surfing activity (S) and vice versa.

- *Converging connections*: For example, the connection $G \rightarrow W \leftarrow M$ in Figure 3.2. Child node W is conditionally dependent on parent nodes G and M. Entering hard evidence at node G will update node W but have no effect on node M. However, if some evidence is already present in node W, then entering information in any of the parent nodes G or M will update the other parent node. Here, soft or hard evidence in node W or any of its descendants, *d-connects* nodes G and M.

  Thus, it can be said that nodes G and M are conditionally dependent if evidence on W or

its descendants is available. This rule tells us that if nothing is known about a common effect of two (or more) causes, then the causes are independent. In other words *Gales* is not an indicator of *Moon*, and vice versa. However, as soon as some evidence is available on a common effect, the causes become dependent. If, for example, we receive some information on the state of *Waves*, then *Gales* and *Moon* become competing explanations for this effect. Thus, receiving information about one of the causes either confirms or dismisses the other one as the cause of *Waves*. Note that even if the initial information about the *Waves* is not reliable (soft evidence), *Gales* and *Moon* still become dependent.

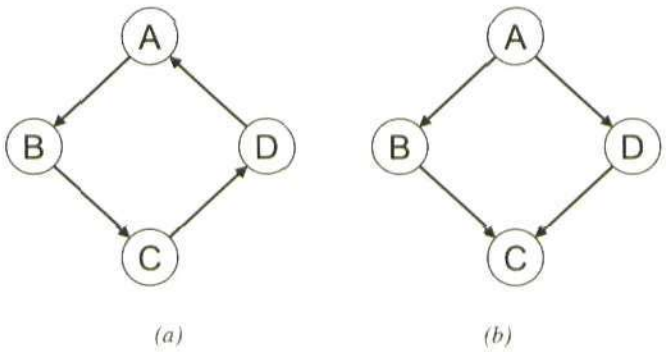
The property of converging connections, where information about the state of a parent node provides an explanation for an observed effect, and hence confirms or dismisses another parent node as the cause of the effect, is often referred to as the *explaining away* effect or as intercausal inference. For example, knowing *Gales* are present strongly suggests these are responsible for the *Waves*, hence explaining away the *Moon* as the cause of the *Waves*.

Critically, one of the fundamental properties of directed acyclic graphs (Bayesian networks) is that for a given node X, the set of its parents, Π_X, *d-separates* this node from all other subsets $\bar{Y}$ with no descendants of X, such that $X \perp\!\!\!\perp \bar{Y} \mid \Pi_X$. In other words, each node in the network is conditionally independent from its non-descendants, given its parents. This allows us to obtain the factorization of the joint probability distribution shown in Equation (3.9), as the following property is satisfied:

$$P(X_i|\Pi_{X_i}, \bar{Y}) = P(X_i|\Pi_{X_i}) \tag{3.11}$$

3.3.2.3 Cycles and acyclic graphs

A *chain* consists of a series of nodes where each successive node in the chain is connected to the previous one by an edge. A *path* is a chain where each connection edge in the chain has the same directionality, i.e. all are serial connections. For example, nodes $M \rightarrow W \rightarrow S$ (Figure 3.2 form a path; while nodes $M \rightarrow W \leftarrow G$ form a chain but not a path. A *cycle* is a path that starts



(a) (b)

Figure 3.3: Bayesian networks can have loops but not cycles. a) A cycle: a path that starts and ends at the same node. b) A loop: a chain with no cycles, where at least one node is visited more than once.

and ends at the same node (Figure 3.3a). A loop, however, is a chain where at least one node is *visited* more than once (i.e. has two or more parents), but has no cycles (Figure 3.3b).

The distinction is important because Bayesian networks by definition have no cycles (acyclic), but can have loops. Bayesian networks with no loops are usually called singly-connected networks, while those with loops are called multiply-connected networks.

3.3.3 Belief propagation

3.3.3.1 *Inference*

Given the structure of the network and the conditional probabilities defining the joint probability distribution (Equation (3.9)), it is possible to analytically compute the marginal probability of each node, in terms of sums over all the possible states of all other nodes in the system i.e. using marginalization, as shown in Equation (3.4). For example, the marginal probability of the variable W can be calculated from the joint probability given by Equation 3.10 as follows,

$$P(W) = \sum_{S}\sum_{F}\sum_{G}\sum_{M} P(S|W) \cdot P(F|W) \cdot P(W|G,M) \cdot P(G) \cdot P(W) \tag{3.12}$$

As can be seen, this computation is impractical, specially for large networks, as the number of terms in the sums grows exponentially with the number of variables. Furthermore, there are

many common intermediate terms in the expressions for the different marginal probabilities, which implies a high redundancy and thus low efficiency in the calculations. Additionally, when new evidence arrives into the network, the effects of the *observed* node modify the marginal probabilities of all other nodes, requiring the whole marginalization process to be repeated for each variable.

Belief propagation, a message-passing algorithm, manages to perform inference in a Bayesian network in a way that grows only linearly with the number of nodes, as it exploits the common intermediate terms that appear in the calculations. In belief propagation the effects of the observation are propagated throughout the network by passing messages between nodes. The final belief, or posterior probability, is computed locally at each node by combining all incoming messages, i.e. evidence from higher and lower levels.

The belief propagation algorithm is not restricted to solving inference problems in Bayesian networks. In fact, a generalized version of the algorithm, also called the sum-product algorithm, can be shown to encompass a number of methods from different disciplines such as physics, digital communications and artificial intelligence. Some of the methods that can be considered particular cases of belief propagation are the forward-backward algorithm, the Viterbi algorithm, decoding algorithms such as turbo-codes, the Kalman filter and the transfer-matrix in physics (Yedidia et al. 2003, Kschischang et al. 2001).

To derive particular instantiations of the belief propagation algorithm it is necessary to consider mathematical scenarios with very specific conditions in each case. For example, the Kalman filter is derived from applying the the generalized belief propagation algorithm to a set of Gaussian random variables that follow certain discrete-time dynamical equations. It is useful to represent the different problems using *factor graphs*, a graph-based language that allows us to represent a set of variables, together with a generic set of functions which relates different subsets of these variables (Yedidia et al. 2003). It has been shown that factor graphs can capture a wide range of mathematical systems, including Markov random fields and Bayesian networks. It is therefore possible to convert any arbitrary Bayesian network into a precisely mathematically equivalent factor graph (and vice versa) and apply the generalized belief propagation algorithm to solve

the inference problem defined by the network (Kschischang et al. 2001).

In this thesis however, I have used Pearl's original belief propagation algorithm applied to Bayesian networks (Pearl 1988). The rationale behind this choice is that, although factor graphs can capture the same phenomena, Bayesian networks provide a more intuitive and explicit account of the causal relations between the variables. I believe this is crucial when modelling hierarchical object recognition in the visual system from the the generative model perspective.

3.3.3.2 Combination of evidence and belief calculation

The aim of belief propagation is to calculate the marginal probability of a variable X given some evidence $\mathbf{e}$. The influence of evidence can propagate to node X either through its parent or through its child nodes, thus evidence can be divided into two subsets such that,

$$\mathbf{e} = \mathbf{e}_X^+ \cup \mathbf{e}_X^- \tag{3.13}$$

$$\mathbf{e}_X^+ \cap \mathbf{e}_X^- = \emptyset \tag{3.14}$$

where $\mathbf{e}_X^+$ represents the evidence *above* node X, and $\mathbf{e}_X^-$ represents the evidence *below* node X. This is shown in Figure 3.4. Similarly, in this section we will use the symbol $\mathbf{e}_{U_i X}^+$ to designate the evidence *above* the link $U_i \rightarrow X$; while $\mathbf{e}_{U_i X}^-$ refers to the evidence *below* the link $U_i \rightarrow X$.

The probability of a node X given some evidence $\mathbf{e}$, i.e. P(x|$\mathbf{e}$), is usually referred to as the posterior probability, $P^*(X)$, or the belief, $Bel(X)$. Taking into account all the above, as well as Bayes rule (Equation (3.8)), we can write,

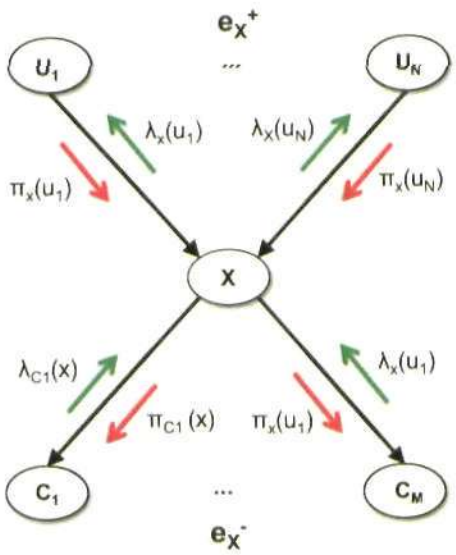


Figure 3.4: Message passing in belief propagation in a Bayesian network. Node X receives all bottom-up messages $\lambda_{C_1}(x),...,\lambda C_M(x)$ from its children, and all top-down messages $\pi_X(u_1),...,\pi_X(u_N)$ from its parents. The belief can then be calculated by combining all bottom-up evidence $\mathbf{e}_X^-$ and top-down evidence $\mathbf{e}_X^+$. Node X generates outgoing messages $\lambda_X(u_1),..,\lambda_X(u_N)$ for its parent nodes, and messages $\pi_{C_1}(x),..,\pi_{C_M}(x)$ for its child nodes.

$$Bel(x) = P^*(x) = P(x|\mathbf{e}) = P(x|\mathbf{e}_X^+, \mathbf{e}_X^-) \tag{3.15}$$

$$= \frac{P(\mathbf{e}_X^-|x, \mathbf{e}_X^+) \cdot P(x|\mathbf{e}_X^+)}{P(\mathbf{e}_X^-|\mathbf{e}_X^+)} \tag{3.16}$$

$$= \alpha \cdot P(\mathbf{e}_X^-|x) \cdot P(x|\mathbf{e}_X^+) \tag{3.17}$$

$$= \alpha \cdot \lambda(x) \cdot \pi(x) \tag{3.18}$$

where $\alpha = [P(\mathbf{e}_X^-|\mathbf{e}_X^+)]^{-1}$, represents a normalization constant; $\lambda(x) = P(\mathbf{e}_X^-|x)$, represents the diagnostic or *retrospective* support that the assertion $X = x$ receives from X's descendants; and $\pi(x) = P(x|\mathbf{e}_X^+)$, represents the causal or *predictive* support that the assertion $X = x$ receives from all non-descendants of X, via X's parents. Thus, the total strength of the belief $X = x$ can be obtained by combining or *fusing* the contribution of bottom-up diagnostic evidence, $\lambda(x)$,

and top-down causal evidence, $\pi(x)$, as shown in Equation (3.18). The λ and π symbols (greek letter for l and p) are chosen because the terms are derived from the likelihood and prior terms, respectively, in the Bayes theorem (Equation (3.8)).

To understand how information from several descendants is combined at node X, we partition the set $\mathbf{e}_X^-$ into disjoint subsets, one for each child of X. For example, for the graph in Figure 3.4, if X is not instantiated, we can write $\mathbf{e}_X^- = \mathbf{e}_{XC_1}^- \cup \mathbf{e}_{XC_2}^- \cup ... \cup \mathbf{e}_{XC_M}^-$, yielding the following expression,

$$\begin{aligned}
\lambda(x) &= P(\mathbf{e}_X^-|x) \\
&= P(\mathbf{e}_{XC_1}^-, ..., \mathbf{e}_{XC_M}^-|x) \\
&= P(\mathbf{e}_{XC_1}^-|x) \cdot ... \cdot P(\mathbf{e}_{XC_M}^-|x) \\
&= \lambda_{C_1}(x) \cdot ... \cdot \lambda_{C_M}(x) \\
&= \prod_{j=1..M} \lambda_{C_j}(x)
\end{aligned} \tag{3.19}$$

where $\lambda_{C_j}(x) = P(\mathbf{e}_{XC_j}^-|x)$, represents the support the assertion $X = x$ receives from the set of nodes *below* the link $X \rightarrow C_j$. In other words, it represents how well the cause x explains the effects observed in the nodes under the C_j. The $\lambda_{C_j}(x)$ terms can be understood as messages that node X receives from its children, which combined via the product rule yield the $\lambda(x)$ function.

Next I will describe how to obtain the $\pi(x)$ function, firstly for a single parent, and secondly for the case with multiple parents (polytrees). Assuming X has a single parent node U and conditioning on the values of U, we get,

$$\begin{aligned}\pi(x) &= P(x|\mathbf{e}_X^+) \\ &= \sum_u P(x|\mathbf{e}_X^+, u) \cdot P(u|\mathbf{e}_X^+) \\ &= \sum_u P(x|u) \cdot P(u|\mathbf{e}_X^+) \\ &= \sum_u P(x|u) \cdot \pi_X(u) \end{aligned} \tag{3.20}$$

where $P(x|u)$ is the conditional probability distribution stored on the link $U \rightarrow X$; and $\pi_X(u) = P(u|\mathbf{e}_X^+)$, represents how probable the value u is, based on causal evidence above the link $U \rightarrow X$. $\pi_X(u)$ can be understood as a message that node X receives from its parent, which, multiplied by the conditional probability function that relates both nodes, yields the $\pi(x)$ function.

In the more general case where multiple parents are present, we assume the evidence can be partitioned such that $\mathbf{e}_X^+ = \mathbf{e}_{U_1X}^+ \cup \mathbf{e}_{U_2X}^+ \cup ... \cup \mathbf{e}_{U_NX}^+$ (Figure 3.4). Therefore,

$$\begin{aligned}\pi(x) &= P(x|\mathbf{e}_X^+) \\ &= P(x|\mathbf{e}_{U_1X}^+, ..., \mathbf{e}_{U_NX}^+) \\ &= \sum_{u_1,...,u_N} P(x|u_1, ..., u_N) \cdot P(u_1, ..., u_N|\mathbf{e}_{U_1X}^+, ..., \mathbf{e}_{U_NX}^+) \\ &= \sum_{u_1,...,u_N} P(x|u_1, ..., u_N) \cdot P(u_1|\mathbf{e}_{U_1X}^+) \cdot ... \cdot P(u_N|\mathbf{e}_{U_NX}^+) \\ &= \sum_{u_1,...,u_N} P(x|u_1, ..., u_N) \cdot \pi_X(u_1) \cdot ... \cdot \pi_X(u_N) \\ &= \sum_{u_1,...,u_N} P(x|u_1, ..., u_N) \cdot \prod_{i=1..N} \pi_X(u_i) \end{aligned} \tag{3.21}$$

where $P(x|u_1, ..., u_N)$ represents the conditional probability distribution that relates node X to

all of its parents; and $\pi_X(u_i)$ represents the support the assertion $U_i = u_i$ receives from the nodes above U_i. $\pi_X(u_i)$ can also be understood as the message X receives from its parent node U_i, which combined with the messages from all other parent nodes, and multiplied by the appropriate values of the conditional probability function, yields the $\pi(x)$ function.

Therefore, as has been demonstrated, the generic node X in Figure 3.4 can calculate its own belief if it receives the messages $\lambda_{C_j}(x)$ from its children and $\pi_X(u_i)$ from its parents. In the rest of this section we will consider how to generate these messages, which allow the evidence to propagate across the network.

3.3.3.3 Bottom-up messages

Taking into account that all nodes in a Bayesian network perform the same operations, we will consider the generic message $\lambda_X(u_i)$, which node X must send to its parent node U_i (Figure 3.4). It is therefore convenient to treat all parents of X, except the one receiving the message, as a common set V, such that $V = \mathbf{U} - U_i = \{U_1, ..., U_{i-1}, U_{i+1}, ..., U_N\}$ as shown in Figure 3.5.

The message $\lambda_X(u_i)$ must take into account all evidence under the link $U_i \rightarrow X$, which includes evidence coming from all other parents of X ($\mathbf{e}^+_{VX} = \bigcup_{k=1..N\setminus i} \mathbf{e}^+_{U_kX}$); and evidence arriving from the descendents of X ($\mathbf{e}^-_X$). Given that X separates $\mathbf{e}^+_{VX}$ from $\mathbf{e}^-_{U_iX}$ and V separates $\mathbf{e}^+_{VX}$ from U_i, we can write,

$$
\begin{aligned}
\lambda_X(u_i) = P(\mathbf{e}^-_{U_iX}|u_i) &= P(\mathbf{e}^+_{VX}, \mathbf{e}^-_X|u_i) \\
\text{[conditioning on } x \text{ and } v\text{]} &= \sum_x \sum_v P(\mathbf{e}^+_{VX}, \mathbf{e}^-_X|u_i, v, x) \cdot P(v, x|u_i) \\
&= \sum_x \sum_v P(\mathbf{e}^-_X, x) \cdot P(\mathbf{e}^+_{VX}, v) \cdot P(v, x|u_i) \\
\text{[applying Bayes theorem]} &= \beta \sum_x \sum_v P(\mathbf{e}^-_X, x) \cdot \frac{P(v|\mathbf{e}^+_{VX})}{P(v)} \cdot P(x|v, u_i) \cdot P(v|u_i) \\
\text{[since } P(v|u_i) = P(v)\text{]} &= \beta \sum_x \sum_v P(\mathbf{e}^-_X, x) \cdot P(v|\mathbf{e}^+_{VX}) \cdot P(x|v, u_i)
\end{aligned}
\tag{3.22}
$$

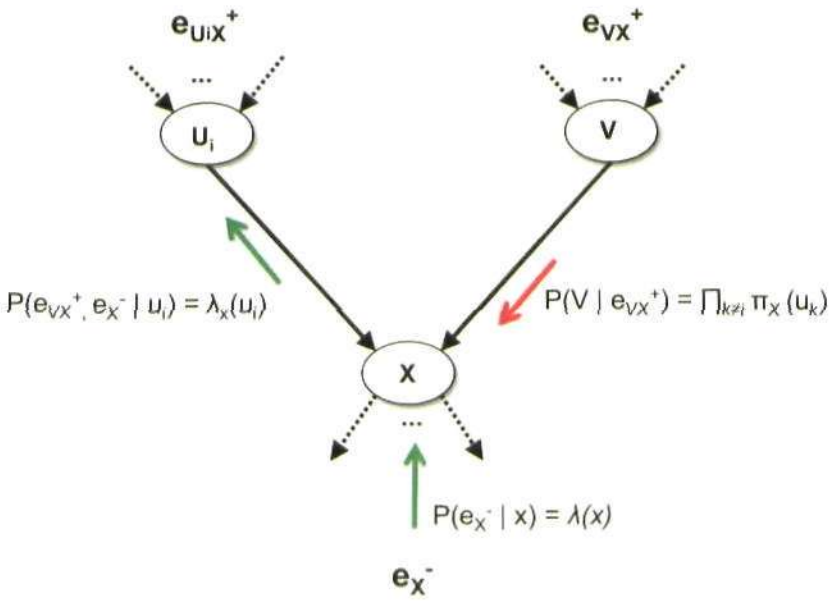


Figure 3.5: Bottom-up λ messages explanatory diagram. The message $\lambda_X(u_i)$ must take into account all evidence under the link $U_i \rightarrow X$, which includes evidence coming from all other parents of X ($\mathbf{e}^+_{VX} = \bigcup_{k=1..N\setminus i} \mathbf{e}^+_{U_kX}$); and evidence arriving from the descendents of X ($\mathbf{e}^-_X$). It is therefore convenient to treat all parents of X, except the one receiving the message, as a common set V, such that $V = \mathbf{U} - U_i = \{U_1,...,U_{i-1},U_{i+1},...,U_N\}$.

After restoring the original meaning of V, such that $P(x|v,u_i) = P(x|u_1,...u_N)$, and

$$P(v|\mathbf{e}^+_{VX}) = \prod_{k=1..N\setminus i} P(u_k|\mathbf{e}^+_{U_kX}) = \prod_{k=1..N\setminus i} \pi_X(u_k) \tag{3.23}$$

the final expression is

$$\lambda_X(u_i) = \beta \sum_x \left[\lambda(x) \sum_{u_1,...,u_N\setminus u_k} P(x|u_1,...u_N) \prod_{k=1..N\setminus i} \pi_X(u_k) \right] \tag{3.24}$$

where β is a normalizing constant, and $\lambda(x)$ is defined in Equation (3.24). Note, in the previous derivation we assume nodes X and V themselves are not instantiated and are therefore not part of the evidence sets $\mathbf{e}^-_{U_iX}$ and $\mathbf{e}^+_{VX}$, respectively.

3.3.3.4 Top-down messages

We now consider the generic message $\pi_{C_j}(x)$ that node X sends to its child node C_j, as shown in Figure 3.4. The body of evidence which must be sent in this message includes all evidence available, except the evidence found in the subnetwork below the link $X \rightarrow C_j$, i.e. $\mathbf{e}^+_{XC_j} = \mathbf{e} - \mathbf{e}^-_{XC_j}$. Therefore, $\pi_{C_j}(x)$ is equivalent to the belief of X (Equation (3.18)) when the evidence $\mathbf{e}^-_{XC_j}$ is suppressed, which can be written as,

$$\begin{aligned} \pi_{C_j}(x) &= \mathbf{e}^+_{XC_j} \\ &= \alpha \cdot \frac{Bel(x)}{\lambda_{C_j}(x)} && (3.25) \\ &= \alpha \prod_{k=1..M\setminus j} \lambda_{C_k}(x) \cdot \pi(x) && (3.26) \end{aligned}$$

where α is a normalizing constant, and $\pi(x)$ is defined in Equation (3.20).

3.3.3.5 Summary of belief propagation rules

Here we sum up the computations performed locally by a node in the generic section of a hierarchical Bayesian network represented in Figure 3.5. Given a node X with parent nodes $U_1,..,U_N$, and a set of child nodes $C_1,..,C_M$, the belief propagation algorithm can be performed in three steps as follows:

1. Node X receives all bottom-up messages $\lambda_{C_1}(x),...,\lambda C_M(x)$ from its children, and all top-down messages $\pi_X(u_1),...,\pi_X(u_N)$ from its parents.

2. Given the fixed conditional probability distribution $P(x|u_1,...,u_N)$ that relates node X to its immediate parents $U_1,...,U_N$, node X can calculate its belief as

$$Bel(x) = \alpha \cdot \lambda(x) \cdot \pi(x) \tag{3.27}$$

$$\lambda(x) = \prod_{j=1..M} \lambda_{C_j}(x) \tag{3.28}$$

$$\pi(x) = \sum_{u_1,...,u_N} P(x|u_1,...,u_N) \cdot \prod_{i=1..N} \pi_X(u_i) \tag{3.29}$$

   where $\lambda(x)$ represents the combination of bottom-up evidence arriving at node X and $\pi(x)$ represents the combination of top-down evidence arriving at node X.

3. Node X generates outgoing messages $\lambda_X(u_1),..,\lambda_X(u_N)$ for its parent nodes, and messages $\pi_{C_1}(x),..,\pi_{C_M}(x)$ for its child nodes, given by the following equations:

$$\lambda_X(u_i) = \beta \sum_x \left[\lambda(x) \sum_{u_1,...,u_N \setminus u_i} P(x|u_1,...u_N) \prod_{k=1..N \setminus i} \pi_X(u_k) \right] \tag{3.30}$$

$$\pi_{C_j}(x) = \alpha \prod_{k=1..M \setminus j} \lambda_{C_k}(x) \cdot \pi(x) = \alpha \cdot \frac{Bel(x)}{\lambda_{C_j}(x)} \tag{3.31}$$

Note the $\lambda_X(u_i)$ message can be sent to node U_i as soon as messages from all other nodes, except node U_i, have been received. Analogously $\pi_{C_j}(x)$ can be sent as soon as all messages, except that arriving from node C_j, have been received.

3.3.3.6 Boundary conditions and evidence nodes

There are four types of nodes that are considered special cases and need to be initialized as follows:

1. *Root nodes:* For a node X without parents, $\pi(x)$ is set equal to the prior probability $P(x)$.

2. *Anticipatory nodes:* For a node X without children, which has not been instantiated, $\lambda(x)$ is set equal to a flat distribution $(1,1,...,1)$, so that $Bel(x)$ is equal to $\pi(x)$.

3. *Evidence nodes:* For a node X that has been instantiated, such that the j-th value of X is observed to be true, $\lambda(x)$ is set equal to $(0,...,0,1,0,...0)$ with 1 at the j-th position. This is usually referred to as hard evidence.

4. *Dummy nodes:* A node X can receive virtual or judgmental evidence from a child dummy node C. In this case the $\lambda(c)$ and $\pi(c)$ do not exist, but instead a $\lambda_C(x)$ message from C to X is generated where $\lambda_C(x) = \beta \cdot P(observation|x)$. The observation can consist of any probability distribution over the states of node X, and is usually referred to as soft evidence.

3.3.3.7 Example of belief propagation with diagnostic evidence

When evidence occurs in the child node and propagates to the parent node, from known effects to unknown causes, this is denoted as diagnostic reasoning or bottom-up recognition. Figure 3.6 shows a scenario, based on the previously described toy example, where evidence about *Surfing* propagates across the network, updating the beliefs in all other nodes.

Note that because variables are binary, the *probability of X* or the *belief of X* refer to the probability of variable X being in the *true* state.

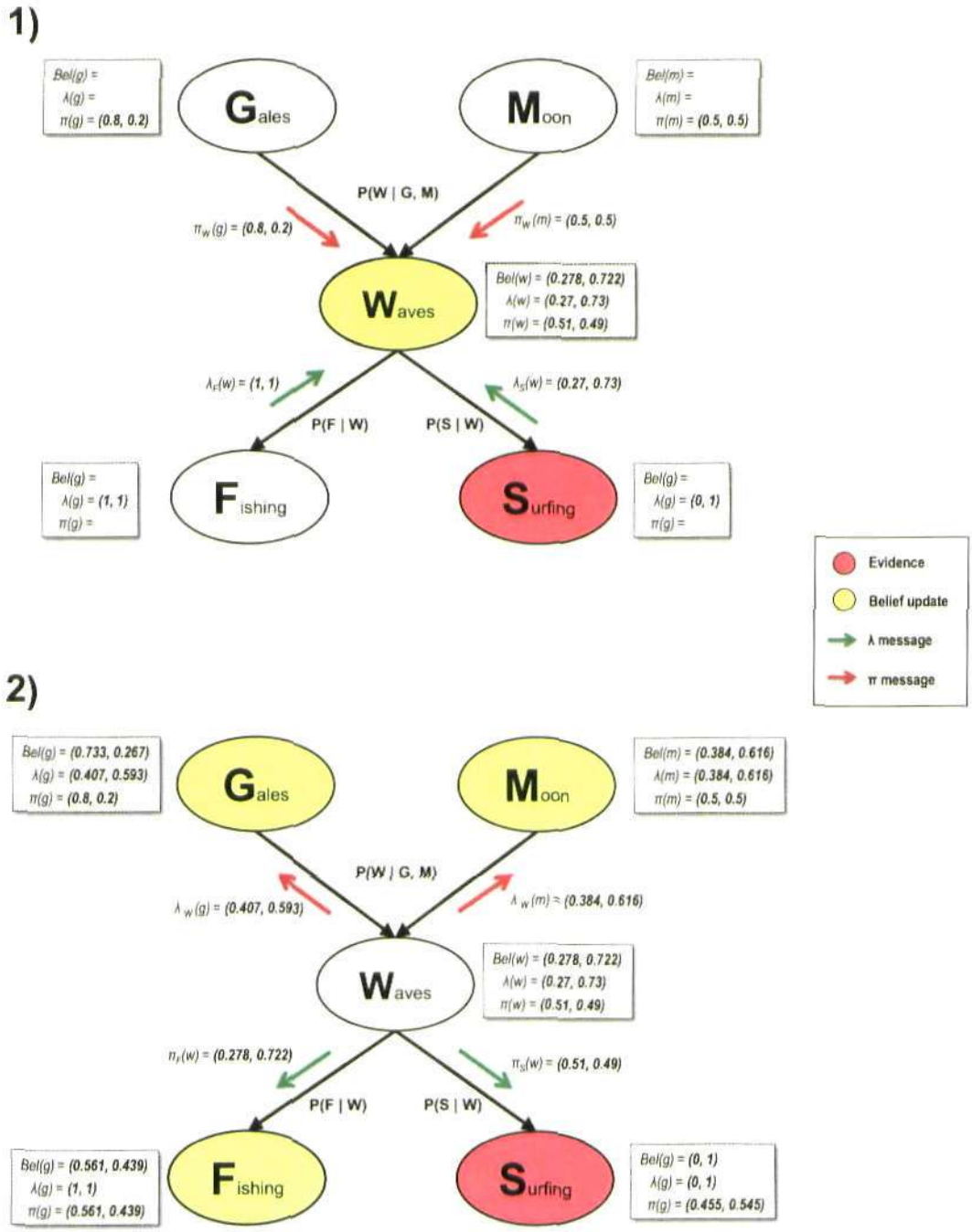


Figure 3.6: For caption see footnote[1].

We assume the following initial conditions: the two root nodes M and G take the value of their prior probabilities; child node F acts as an anticipatory node, thus maintaining a flat distribution, and node S is an evidence node with a distribution $(0, 1)$, i.e. hard evidence indicating that there is definitely *Surfing* activity. Given this information and the conditional probability tables provided in Figure 3.6, all the nodes are ready to send messages to the intermediate node W. The initial conditions, and the subsequent generated messages following Equations (3.30) and (3.31), are shown below:

[1]Caption for Figure 3.6. Example of belief propagation with diagnostic evidence. The scenario, based on the previously described toy example, shows how evidence about *Surfing* propagates across the network, updating the beliefs in all other nodes. In the first step the two root nodes and the two leaf nodes send messages to the intermediate node W, which multiplies the combined top-down evidence, $\pi(w)$ and bottom-up evidence, $\lambda(w)$, to obtain its belief. Step two shows the outgoing messages from node W to its child and parent nodes. The key property here is that the outgoing messages take into account all evidence except that which originated from the destination node. For example the message from W to G, $\lambda_W(g)$, doesn't take into account the prior information conveyed through the incoming message $\pi_W(g)$. The belief, or posterior probability, of both parent nodes is higher than their original *prior* probability. Node F also updates its belief according to the new incoming message, showing a lower probability. Intuitively, evidence of surfing activity suggests the presence of big waves, which in turn suggests the presence of gales and/or the moon as the generating causes. At the same time, the presence of big waves suggests fishing activity is less likely to be present. See the text for a detailed step by step description of the mathematical operations.

$$\lambda(f) = (1,1);\ \ \lambda(s) = (0,1);\ \ \pi(g) = (0.8,0.2);\ \ \pi(m) = (0.5,0.5)$$

$$\begin{aligned}\lambda_F(w) &= \beta \cdot \sum_f \lambda(f) \cdot P(f|w) \\ &= \beta \cdot (1 \cdot 0.2 + 1 \cdot 0.8, 1 \cdot 0.7 + 1 \cdot 0.3) = (1,1)\end{aligned}$$

$$\begin{aligned}\lambda_S(w) &= \beta \cdot \sum_s \lambda(f) \cdot P(s|w) \\ &= (0 \cdot 0.7 + 1 \cdot 0.3, 0 \cdot 0.2 + 1 \cdot 0.8) \\ &= \beta \cdot (0.3,0.8) = (0.27,0.73)\end{aligned}$$

$$\pi_W(g) = \pi(g) = (0.8,0.2)$$

$$\pi_W(m) = \pi(m) = (0.5,0.5)$$

Once node W has received all incoming messages, it can generate $\lambda(w)$, which combines all bottom-up evidence, and $\pi(w)$, which combines all top-down evidence (in this case the priors). The final belief can be obtained by multiplying together these two factors, thus obtaining the marginal probability of W given all the information available in the network. Following Equations (3.27), (3.28) and (3.29), we obtain

$$\begin{aligned}\lambda(w) &= \lambda_F(w) \cdot \lambda_S(w) \\ &= (1,1) \cdot (0.27,0.73) = (0.27,0.73)\end{aligned}$$

$$\pi(w) = \sum_{g,m} P(w|g,m) \cdot \pi_W(g) \cdot \pi_W(m)$$

$$\begin{aligned}\pi(w=0) &= (P(w=0|g=0,m=0) \cdot \pi_W(g=0) \cdot \pi_W(m=0)) + \ldots \\ &\quad + (P(w=0|g=1,m=0) \cdot \pi_W(g=1) \cdot \pi_W(m=0)) + \ldots \\ &\quad + (P(w=0|g=0,m=1) \cdot \pi_W(g=0) \cdot \pi_W(m=1)) + \ldots \\ &\quad + (P(w=0|g=1,m=1) \cdot \pi_W(g=1) \cdot \pi_W(m=1)) \\ &= (0.9 \cdot 0.8 \cdot 0.5) + (0.2 \cdot 0.2 \cdot 0.5) + (0.3 \cdot 0.8 \cdot 0.5) + (0.9 \cdot 0.2 \cdot 0.5) = 0.51\end{aligned}$$

$$\pi(w=1) = (0.1 \cdot 0.8 \cdot 0.5) + (0.8 \cdot 0.2 \cdot 0.5) + (0.7 \cdot 0.8 \cdot 0.5) + (0.9 \cdot 0.2 \cdot 0.5) = 0.49$$

$$\pi(w) = (0.51,0.49)$$

$$\begin{aligned}Bel(w) &= \alpha \cdot \lambda(w) \cdot \pi(w) \\ &= \alpha(0.27,0.73) \cdot (0.51,0.49) = \alpha \cdot (0.138,0.358) \\ &= (0.278,0.722)\end{aligned}$$

In this case, the evidence in *Surfing* yields a value of $\lambda(w)$ that suggests there is a high probability of *Waves* (0.73). The top-down prior information $\pi(w)$ is practically a flat distribution (0.51,0.49), i.e. doesn't add any information, thus the resulting belief suggests there is a high probability of *Waves* (0.722). This is shown graphically in the top diagram of Figure 3.6.

The next step is to generate the outgoing messages from node W to its child and parent nodes. The key property here is that the outgoing messages take into account all evidence except that which originated from the destination node. For example, the message from W to G, $\lambda_W(g)$, doesn't take into account the prior information conveyed through the incoming message $\pi_W(g)$. Given Equations (3.30) and (3.31), the resulting expressions are

$$\lambda_W(g) = \beta \cdot \sum_w \lambda(w) \sum_m P(w|g,m) \cdot \pi_W(m)$$

$$\begin{aligned}\lambda_W(g=0) &= \lambda(w=0) \cdot (P(w=0|g=0,m=0) \cdot \pi_W(m=0) + ... \\ &\quad + P(w=0|g=0,m=1) \cdot \pi_W(m=1)) + ... \\ &\quad + \lambda(w=1) \cdot (P(w=1|g=0,m=0) \cdot \pi_W(m=0) + ... \\ &\quad + P(w=1|g=0,m=1) \cdot \pi_W(m=1)) \\ &= 0.27 \cdot (0.9 \cdot 0.5 + 0.3 \cdot 0.5) + 0.73 \cdot (0.1 \cdot 0.5 + 0.7 \cdot 0.5) = 0.454\end{aligned}$$

$$\lambda_W(g=1) = 0.27 \cdot (0.2 \cdot 0.5 + 0.1 \cdot 0.5) + 0.73 \cdot (0.8 \cdot 0.5 + 0.9 \cdot 0.5) = 0.661$$

$$\lambda_W(g) = \beta \cdot (0.454, 0.661) = (0.407, 0.593)$$

$$\lambda_W(m) = \beta \cdot \sum_w \lambda(w) \sum_m P(w|g,m) \cdot \pi_W(g)$$

$$\lambda_W(m=0) = 0.27 \cdot (0.9 \cdot 0.8 + 0.2 \cdot 0.2) + 0.73 \cdot (0.1 \cdot 0.8 + 0.8 \cdot 0.2) = 0.38$$

$$\lambda_W(m=1) = 0.27 \cdot (0.3 \cdot 0.8 + 0.1 \cdot 0.2) + 0.73 \cdot (0.7 \cdot 0.8 + 0.9 \cdot 0.2) = 0.61$$

$$\lambda_W(m) = \beta \cdot (0.38, 0.61) = (0.384, 0.616)$$

$$\begin{aligned}\pi_F(w) &= \beta \cdot \lambda_S(w) \cdot \pi(w) \\ &= \beta \cdot (0.27, 0.73) \cdot (0.51, 0.49) = \beta(0.138, 0.358) = (0.278, 0.722)\end{aligned}$$

$$\begin{aligned}\pi_S(w) &= \beta \cdot \lambda_F(w) \cdot \pi(w) \\ &= \beta \cdot (1,1) \cdot (0.51, 0.49) = (0.51, 0.49)\end{aligned}$$

Note how the bottom-up messages, $\lambda_W(g)$ and $\lambda_W(m)$, take into account, not only the bottom-up evidence, but also the prior probability conveyed by the parent node to which the message is not destined. This is sometimes referred to as *horizontal* or *sideways* interactions between parent nodes with a common successor, and results in the *explaining away* effect. Analogously, the top-

down messages, $\pi_F(w)$ and $\pi_S(w)$, convey information about the top-down prior information together with evidence arriving from the non-recipient child node. Once the above messages reach their destination nodes, it is possible to calculate the Belief in each of the periphery nodes, as shown in the bottom diagram of Figure 3.6,

$$\begin{aligned} Bel(g) &= \alpha \cdot \lambda(g) \cdot \pi(g) = \alpha \cdot \lambda_W(g) \cdot \pi(g) \\ &= \alpha \cdot (0.407, 0.593) \cdot (0.8, 0.2) = \alpha(0.326, 0.119) = (0.733, 0.267) \end{aligned}$$

$$\begin{aligned} Bel(m) &= \alpha \cdot \lambda(m) \cdot \pi(m) = \alpha \cdot \lambda_W(m) \cdot \pi(m) \\ &= \alpha \cdot (0.384, 0.616) \cdot (0.5, 0.5) = \alpha(0.192, 0.308) = (0.384, 0.616) \end{aligned}$$

$$\begin{aligned} Bel(f) &= \alpha \cdot \lambda(f) \cdot \pi(f) = \alpha \cdot \lambda(f) \cdot \sum_w P(f|w) \cdot \pi_F(w) \\ &= \cdot(1,1) \cdot (0.2 \cdot 0.278 + 0.7 \cdot 0.722, 0.8 \cdot 0.278 + 0.3 \cdot 0.722) \\ &= (1,1) \cdot (0.561, 0.439) = (0.561, 0.439) \end{aligned}$$

$$\begin{aligned} Bel(s) &= \alpha \cdot \lambda(s) \cdot \pi(s) = \alpha \cdot \lambda(s) \cdot \sum_w P(s|w) \cdot \pi_S(w) \\ &= \cdot(0,1) \cdot (0.7 \cdot 0.51 + 0.2 \cdot 0.49, 0.3 \cdot 0.51 + 0.8 \cdot 0.49) \\ &= (0,1) \cdot (0.455, 0.545) = (0,1) \end{aligned}$$

Overall, the evidence in S has propagated across the network updating the beliefs of all other variables. First, the evidence arrives at node W, increasing its belief. Node W then sends messages to both parent nodes which show a higher belief, or posterior probability, than the original *prior* probability. Node W also sends a message to node F, which decreases its belief accordingly. Intuitively, evidence of surfing activity suggests the presence of big waves, which in turn suggest the presence of gales and/or the moon as the generating causes. At the same time, the presence of big waves suggests fishing activity is less likely to be present.

Note that node W also sends a message back to the evidence node S, but in this case the resulting belief is equal to the initial evidence and is not affected by the message, i.e. $Bel(s) = \lambda(s) = (0,1)$. This is because the type of evidence was *hard evidence*. If instead, *soft evidence* was used (e.g. $\lambda(s) = (0.1, 0.9)$), the top-down message $\pi_S(w)$ would be able to modify the belief in S. This can be understood if we consider that soft evidence contains a certain degree of uncertainty, and is therefore susceptible to being modulated (confirmed or contradicted) by other information in the network, while hard evidence is assumed to be irrefutable fact.

3.3.3.8 Example of belief propagation with diagnostic evidence and causal evidence (explaining away)

In this scenario there is both bottom-up and top-down evidence. When evidence is propagated from a parent to a child node, from known causes to unknown effects, this is called causal reasoning or top-down prediction. The main purpose of this scenario is to illustrate the *explaining away* effect. For clarity, we omit the detailed numerical calculations for the beliefs and messages, as the reader can easily follow the example using Figure 3.7, which shows all the relevant resulting values. These have been obtained using the same Equations ((3.27)-(3.31)) and procedure described in the previous example.

In this example, the prior probability of *Gales* is set equal to hard evidence asserting the *true* state of the variable. Consequently, top-down evidence from the high-level cause *Gales* is combined with bottom-up evidence from the low-level effect *Surfing*. This is sometimes referred to as *data fusion*. The resulting belief in *Waves* is higher than in the previous scenario (0.87 vs. 0.722). This makes sense, as the variable *Waves* is now receiving positive evidence not only from the child node *Surfing*, but also from the parent node *Gales*, providing further support for the belief *Waves=true*.

This in turn also leads to an update in the belief of *Fishing*, indicating its value is now even lower than for the previous scenario (0.365 vs. 0.439). This is a direct consequence of the probability of *Waves* being higher due to the new evidence introduced in the variable *Gales*. In other words, knowing that the moon is in a state which is likely to generate big waves reduces the chances of fishing activity.

However, the most interesting effect happens at the node *Moon*, where its belief shows a reduction with respect to the previous scenario (0.524 vs. 0616). This is a consequence of the *increased probability of Gales, which suggests it is the cause responsible of Waves, thus explaining away* the *Moon* cause. In other words, once there is an explanation for *Waves*, namely *Gales*, the probability of alternative explanations, such as *Moon*, is reduced. Note the probability of *Moon* is still relatively high, as according to the conditional probability table $P(W|G,M)$, both high-level causes can coexist, and in fact, when both are present, the conditional probability of *Waves* is higher.

3.3.3.9 Example of belief propagation with no evidence

This example serves to illustrate how belief propagation operates when there is no evidence available. All the resulting beliefs and the flow of messages are depicted in detail in Figure 3.8. Strictly speaking the resulting beliefs are not the posterior probabilities, $P(x|\mathbf{e})$, as there is no evidence available. Instead they represent the marginal probabilities of the variables when the network is in an initial equilibrium state before presenting any evidence. Therefore it is also useful to compare the resulting beliefs in the network in equilibrium with those when there is evidence, to obtain a better understanding of the effects of evidence propagation.

When all the bottom-up λ messages received by a node show flat distributions (i.e. no evidence below), inevitably all the λ messages sent to its parents will also show flat distributions, regardless of the incoming π messages. This is the case of node W in Figure 3.8. The prior probability (or evidence) at the top causal nodes G and M does not influence the other causal node, until their common child W gathers some diagnostic evidence. This reflects the *d-separation* con-

[2]Caption for Figure 3.7. Example of belief propagation with diagnostic evidence and causal evidence (explaining away). The prior probability of *Gales* is set equal to hard evidence asserting the *true* state of the variable. Consequently, top-down evidence from the high-level cause *Gales* is combined with bottom-up evidence from the low-level effect *Surfing*. This is sometimes referred to as *data fusion*. The resulting belief in *Waves* is higher than in the previous scenario (0.87 vs. 0.722). This makes sense, as the variable *Waves* is now receiving positive evidence not only from the child node *Surfing*, but also from the parent node *Gales*, providing further support for the belief *Waves=true*. This in turn also leads to an update in the belief of *Fishing*, indicating its value is now even lower than for the previous scenario (0.365 vs. 0.439). This is a direct consequence of the probability of *Waves* being higher due to the new evidence introduced in the variable *Gales*. In other words, knowing that the moon is in a state which is likely to generate big waves, reduces the chances of fishing activity. However, the most interesting effect happens at the node *Moon*, where its belief shows a reduction with respect to the previous scenario (0.524 vs. 0616). This is a consequence of the increased probability of *Gales*, which suggests it is the cause responsible of *Waves*, thus *explaining away* the *Moon* cause. In other words, once there is an explanation for *Waves*, namely *Gales*, the probability of alternative explanations, such as *Moon*, is reduced.

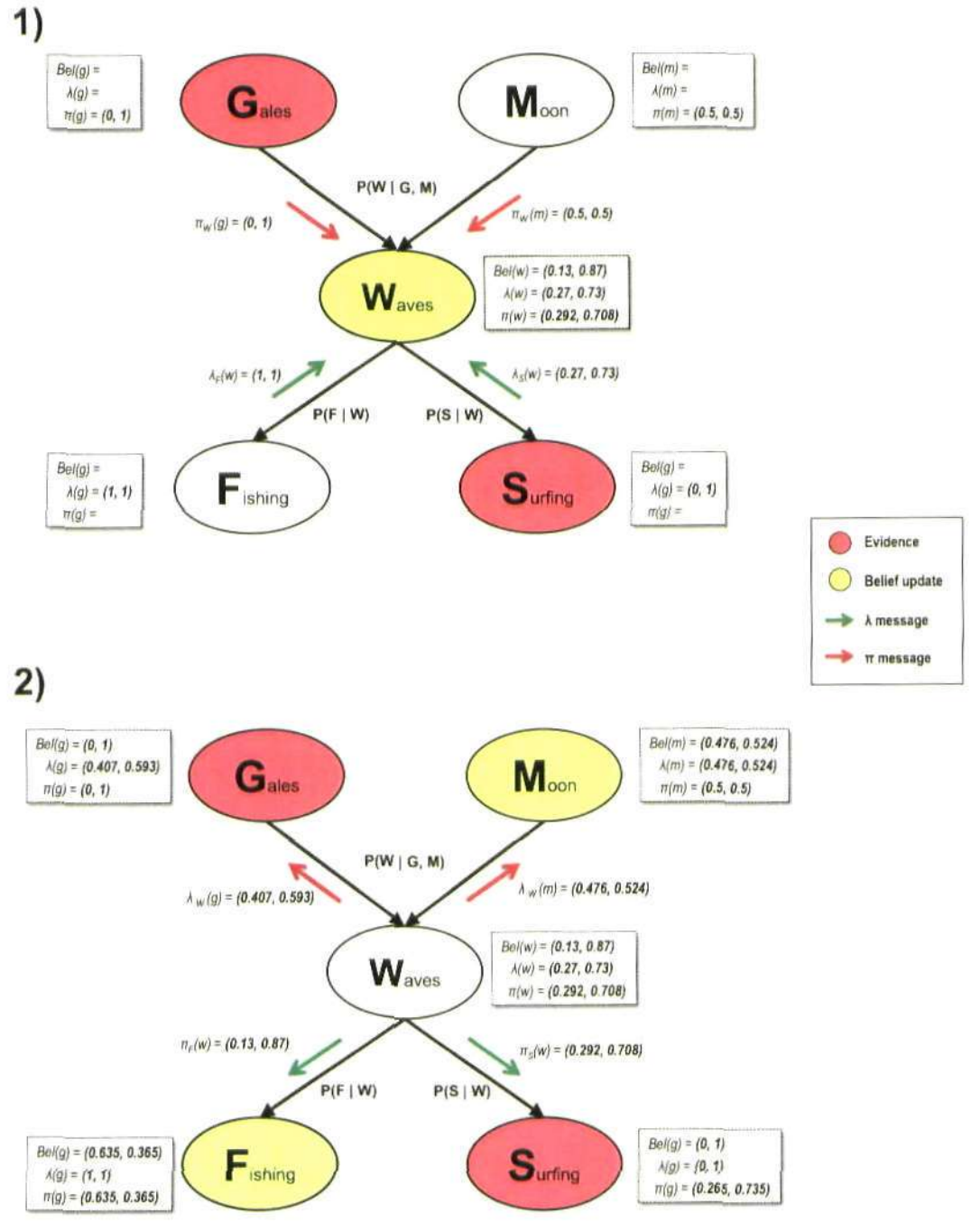


Figure 3.7: For caption see footnote[2].

dition established by converging nodes, described in Section 3.3.2, and matches our intuition regarding multiple causes. Without any information about the state of the *Waves*, data on the state of *Gales* should not influence the state of *Moon*, as they are conditionally independent causes.

3.3.3.10 Example of belief propagation with tree structure

The previous examples are all based on a network with the same simple structure: a central node with two parents and two children node. Although this type of structure serves to demonstrate the main concepts behind belief propagation, it does not capture an interesting effect of tree-structured networks, which is therefore described in this subsection.

In this case the network has three levels organized in a tree structure as shown in Figure 3.9. In the first step, evidence propagates from two of the child nodes in the lower level, leading to the update of the belief in the intermediate nodes. In the second step, the belief at the top level is updated, together with the belief of the lower-lever child nodes that hadn't been instantiated.

The crucial process occurs in step three when a message is sent downward from the top node. Note this didn't happen in the network of the previous examples, where the propagation ended once the message reached the top nodes. The reason is that in this case the top node receives messages from the two intermediate child nodes (the left and the right branches of the tree), and therefore it must generate a top-down message for each node conveying the evidence collected from the other node. In other words the evidence from the left branch must be propagated to the nodes in the right branch and vice versa. This is depicted graphically in steps three and four.

In the original example, an equivalent flow of evidence would happen, for example, if node *Gales* had a second child node, such as *Fallen trees*. Evidence originating in the node *Surfing* would propagate up the node *Waves* to the root node *Gales* and back down the opposite branch,

[3]Caption for Figure 3.8. Example of belief propagation with no evidence. When all the bottom-up λ messages received by a node show flat distributions, as is the case of node W, inevitably all the λ messages sent to its parent nodes will also show flat distributions, regardless of the incoming π messages. The prior probability (or evidence) at the top causal nodes G and M does not influence the other causal node, until their common child W gathers some diagnostic evidence. This reflects the *d-separation* condition established by converging nodes, described in Section 3.3.2, and matches our intuition regarding multiple causes. Without any information about the state of the *Waves*, data on the state of *Gales* should not influence the state of *Moon*, as they are conditionally independent causes.

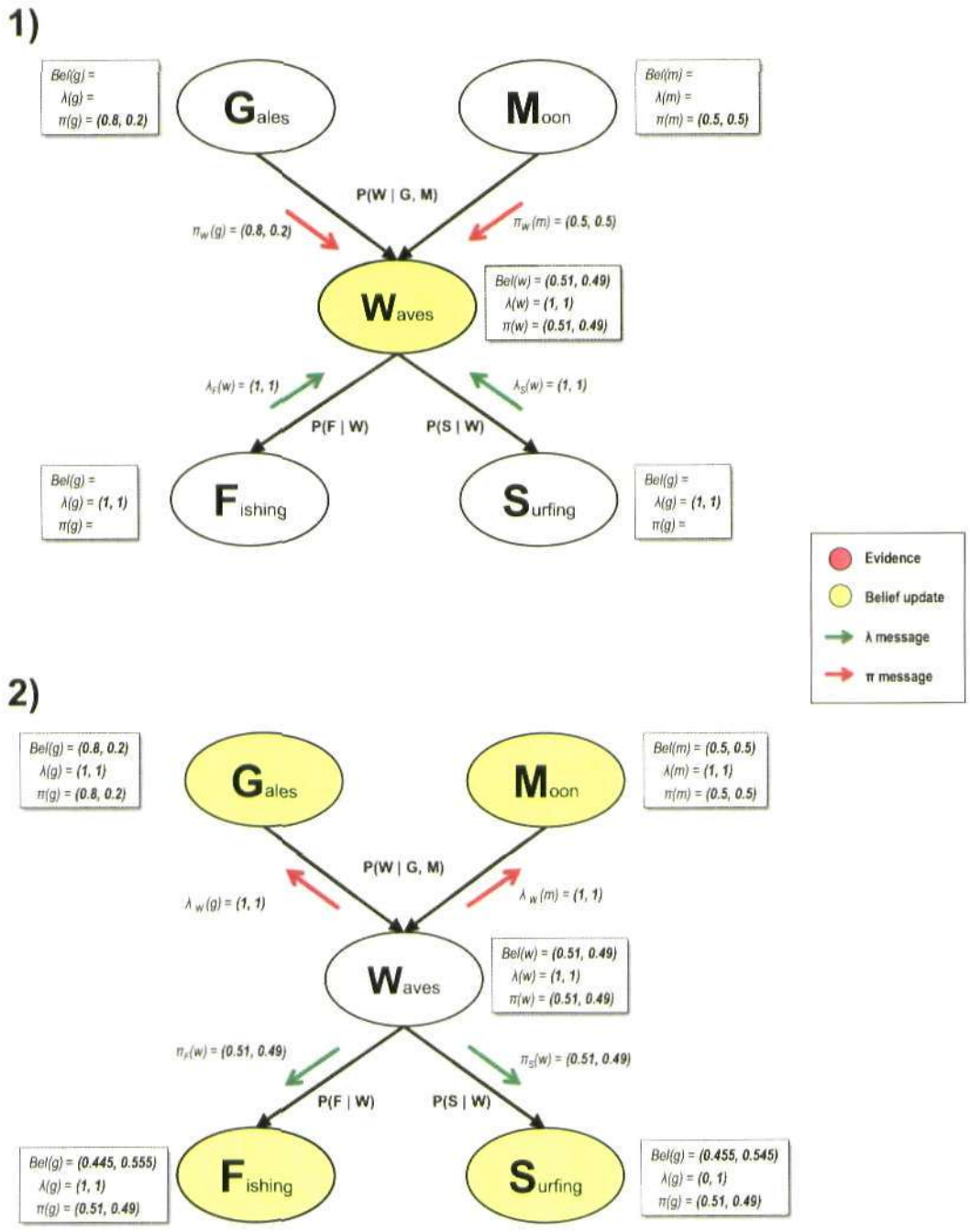


Figure 3.8: For caption see footnote[3].

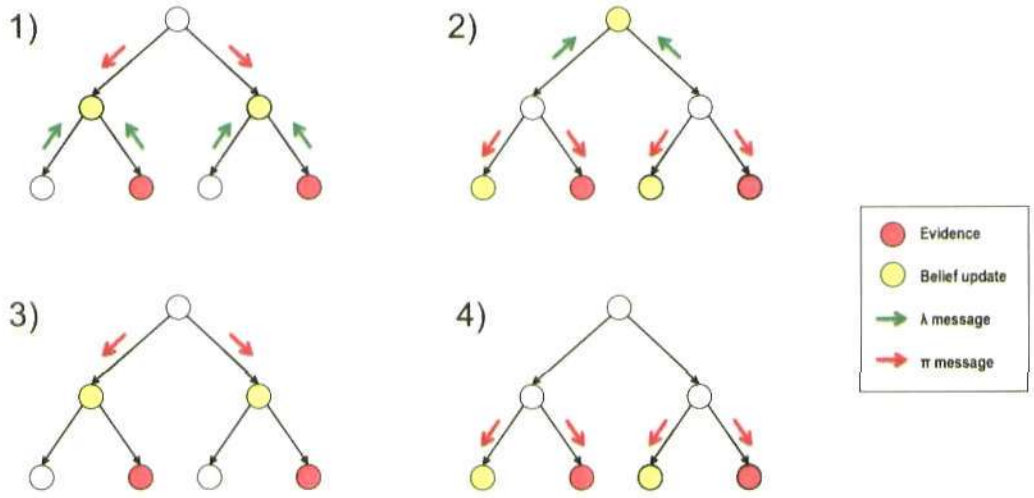


Figure 3.9: Example of belief propagation in a tree-structured network. The network has three levels organized in a tree structure. In the first step evidence propagates from two of the child nodes in the lower level, leading to the update of the belief in the intermediate nodes. In the second step, the belief at the top level is updated, together with the belief at the lower-lever child nodes that hadn't been instantiated. The crucial process occurs in step three when a message is sent downward from the top node. Note this didn't happen in the network of the previous examples, where the propagation ended once the message reached the top nodes. The reason is that in this case the top node receives messages from the two intermediate child nodes (the left and the right branches of the tree), and therefore it must generate a top-down message for each node conveying the evidence collected from the other node. In other words the evidence from the left branch must be propagated to the nodes in the right branch and vice versa. This is shown in steps three and four.

updating the belief of *Fallen trees*. Analogously, evidence originating from the node *Fallen trees* would update the belief of all the nodes in the opposite branch, by flowing up to node *Gales* and down through node *Waves* to nodes *Surfing* and *Fishing*.

We can therefore distinguish between three types of networks. The first two fall into the category of singly-connected networks, those without loops, and the number of steps required to obtain the exact beliefs of all nodes is proportional to the diameter of the network. In singly-connected tree networks (no loops with one parent per node), evidence propagates from the leaf nodes to the root node and back down again (Figure 3.9). This happens because outgoing λ messages do not take into account the π message received from the parent node. In singly-connected polytrees (no loops with multiple parents), there is a single bottom-up top-down

pass for each branch, such that for every connecting arc in the network only one λ and one π messages are required to effectively obtain the exact beliefs (e.g. Figure 3.6).

Note that for a singly-connected network to have nodes with multiple parents, these parents must not be interconnected, otherwise it becomes the third type of network, i.e. a multiply-connected network. In these networks the number of steps required is not fixed as messages circulate indefinitely. Thus, messages from intermediate nodes are typically initialized to a flat distribution and propagate upwards and downwards simultaneously from the first time-step onwards. Feedback in multiply connected networks is described in more detail in Section 3.3.5.

In singly-connected networks, although messages from the root and intermediate nodes could be initialized to a flat distribution and propagated from the first time step, these would just generate temporal beliefs that would not affect the final exact belief. To avoid these extra calculations, belief propagation dictates that nodes only generate output messages once they have received all the required incoming messages. This means the π messages (red arrows) in steps 1 and 2 of Figure 3.9 are redundant, i.e. they don't contribute to the final belief. For this reason, in singly-connected networks, belief propagation can be argued to occur in a single bottom-up and a top-down pass. Another important property of this type of network is that the message propagation scheme can be implemented asynchronously, in other words, it does not require any particular order to provide the correct beliefs.

3.3.4 Combining messages from multiple parents

In discrete Bayesian networks, the conditional probability table (CPT) which relates states of the parent nodes to those of a child node includes entries for all possible combinations of the child and parent node states. Given a node X with k_X states, and its parent nodes $U_1, ..., U_N$ with k_U states each, the number of entries in the CPT is equal to $k_X \cdot k_N^N$. This means the number of entries is exponential to the number of parents, such that even for relatively moderate dimensions ($k_X = k_U = 4, N = 8$), the size of the CPT becomes large and unmanageable (262,144 entries).

Not only does the storage space increase exponentially with the number of parents, but so does the computation time required to compute the belief and the messages at node X. Additionally, learning all the values of the CPT can be problematic as the training data may not include all

combinations of parent node states, and even if it does, large quantities of data are required to avoid overfitting. If the Bayesian network is based on expert's opinions, such as in medical expert systems, it is also usually infeasible to consider all parental state combinations.

In this subsection I present two methods which try to solve this problem by generating CPTs using a number of parameters that is linear to the number of parent nodes. For the model presented in this thesis, these approximations to the CPTs are crucial as the number of parents and states per node is very high.

3.3.4.1 The Noisy-OR gate

This method assumes each of the N parent nodes U_i is sufficient to cause X in absence of other parent nodes, and their ability to cause X is independent of the presence of other causes. For example, the presence of *Gales* by itself is sufficient to cause *Waves*, independently of the presence of the cause *Moon*. This is equivalent to saying that the exceptions or *inhibitors* which may prevent the different parent nodes from causing the effect are independent of each other, denoted as (*exception independence*). For example, an exception that might prevent *Gales* from causing *Waves* is the wind direction (although it does not affect the *Moon*'s ability to cause *Waves)*; while an exception which might prevent *Moon* from causing *Waves* is shallow water depth (although it does not affect the *Gales*'s ability to cause *Waves*).

For binary variables this means the entire conditional probability distribution can be specified with only N parameters $p_1, ..., p_N$, where p_i represents the probability that effect X will be true if the cause U_i is present and all other causes $U_j, i \neq j$, are absent. The system can also be described using the inhibitor probabilities $q_1, ..., q_N$, where q_i denotes the probability that the inhibitor or exception for cause U_i is present. Therefore, we can write,

$$p_i = 1 - q_i = P(x|u_1 = 0, u_2 = 0, ..., u_i = 1, ..., u_{N-1} = 0, u_N = 0) \tag{3.32}$$

If T_U represents the indices of the parent nodes which contain *TRUE* values, $T_U = i : u_i = 1$, then we can derive the complete CPT of X given its predecessors $U_1, ..., U_N$ as,

$$P(x=1|u_1,...,u_N) = 1 - \prod_{i \in T_U} (1-p_i) = 1 - \prod_{i \in T_U} q_i \tag{3.33}$$

For the example described in the Section 3.3.2, if we assume $p_{Gales} = 0.8$ and $p_{Moon} = 0.7$, the CPT value for *Waves=1* in the case when both causes are true can be obtained as follows,

$$P(w=1|g=1,m=1) = 1 - (1-p_{Gales}) \cdot (1-p_{Moon}) = 1 - 0.2 \cdot 0.3 = 0.94$$

which yields a value consistent with the value that was intuitively generated for the CPT of the original example. This serves to illustrate the concept behind the Noisy-OR gate, although obviously in this case it does not make sense to use the method as the number of parent nodes and states is very small.

The Noisy-OR method was originally described for binary variables (Pearl 1988), although it was later extended to variables with multiple states (Srinivas 1993, Diez 1993). However, the main limitation of this method is that it requires *graded* variables to work (Diez 1993), variables whose states can be ordered from lower to higher. The speed of the wind is an example of a graded variable. However, a variable whose states are different object categories is not. The model has also been extended to include different weights for each of the parent links (Kokkonen et al. 2005), such that the relative contribution of each of them can be modulated according to some learned or hard-wired criteria.

The Noisy-OR model describes how causes interact disjunctively. Other canonical models describe different types of parental interaction, such as the Noisy-AND model which describes the conjunctive interaction of causes. For more details on the Noisy-OR and other canonical models see Pearl (1988) and Diez (1993).

3.3.4.2 Weighted sum based on compatible parental configurations

An alternative method to the Noisy-OR is that proposed by Das (2004), based on the concept of *compatible parental configurations*. The rationale behind it is to facilitate the acquisition of probabilistic knowledge, when this is obtained from human experts, by reducing the number

of questions they need to answer. Instead of asking a question for each of the combinations of parental states, the model assumes that for each state of each parent node, the rest of the parents are in a *compatible* or most-likely state.

Previous models, such as the Noisy-OR, are constrained by the assumption that parents act *independently without synergy. This means parents individually influence the child and that* there are negligible cross-interactions between individual parent-to-child influences, such as in the *Waves* example.

However, this method is derived for a different class of Bayesian networks in which there necessarily exists a coherent frame of knowledge where the effect is a result of the interactions between the parents. An excellent example is provided by Das (2004), where the *efficiency* (E) of a company is affected by three parent nodes: *personnel morale* (PM), *personnel training* (PT) and *managerial expertise* (ME), with states *very-low*, *low*, *average*, *high*, *very-high*. Clearly these causes are related, such that a possible compatible parental configuration when the *personnel morale* is *high* is $\{Comp(PM = high)\} = \{PM = high, PT = high, ME = high\}$. This means when the *personnel morale* is *high*, it is also likely that the *personnel training* and the *managerial expertise* are *high*.

More formally, given a node X with a set of parents $U_1, ..., U_N$, the state $U_j = u_j$ is *compatible* with the state $U_i = u_i$, if according to the expert's mental model the state $U_j = u_j$ is most likely to coexist with the state $U_i = u_i$. Let $\{Comp(U_i = u_i)\}$ denote the *compatible parental configuration* where U_i is in the state u_i and the rest of the parents are in states compatible with $U_i = u_i$.

For each compatible parental configuration it is now possible to calculate the conditional probability distribution over the states of the child node X in the form

$$P(X = 0|\{Comp(U_i = u_i)\}), P(X = 1|\{Comp(U_i = u_i)\}), ..., P(X = k_X|\{Comp(U_i = u_i)\})$$

for $i = 1..N$ and $u_i = 1..k_{U_i}$, where k_X is the number of states of X, and k_{U_i} is the number of states of parent node U_i.

Therefore the link is defined using $k_{U_1} + k_{U_2} + ... + k_{U_N}$ probability distributions over X for the different compatible parental configuration. Note these grow linearly with the number of parents. Given a set of weights $w_1, ..., w_N$, which quantify the relative strength of the parent nodes' influence on the child node, the entries of the CPT can be generated using the following weighted sum expression,

$$P(x|u_1, ..., u_N) = \sum_{i=1..N} w_i \cdot P(x|\{Comp(U_i = u_i)\}) \tag{3.34}$$

It is important to stress that $\{Comp(U_i = u_i)\}$ is a parental configuration in the mental model of the expert where he has chosen to focus on the state u_i of parent U_i, while the rest of the states of the parents are perceived in his judgement to be in compatible states with u_i. This helps the expert to simplify his mental model in order to judge the possible effect. It does not mean that compatible parental configurations are the only ones to be found in reality, but these are assumed to be more *common* or *normal*.

The method described here proposes combining the probability distributions of X given compatible parental configurations, to calculate the states of X given *incompatible*, or less common, parental configurations, by using the weighted sum expression in Equation (3.34). This can be understood as a kind of interpolation mechanism that exploits the known data points. Das (2004) makes use of information geometry to demonstrate how these weighted sums capture the experts' judgemental strategy. The method is being employed to design strategic military applications for the Australian Department of Defence.

Although the method was derived for populating CPTs using human experts, theoretically it can be extended to systems that obtain their information using training data with supervised learning methods. One such domain is hierarchical object recognition, where, due to the great overlap between receptive fields, parent nodes show contextual interdependency and can therefore exploit this technique. This is discussed further in Chapter 5, where a toy example is used to illustrate the concept.

3.3.5 Networks with loops and inference methods

A loop is a chain where at least one node is *visited* more than once, as described in Section 3.3.2 and illustrated in Figure 3.3b. Loops are very common in Bayesian networks which try to model real-world data. The belief propagation equations described for singly connected networks are not correct for multiply connected networks (those with loops). The reason is that the equations are based on the assumption that all parents of a node X are mutually independent as long as none of their common descendants are instantiated. This assumption is no longer valid in networks with loops, where some of the parents of X will share a common ancestor.

Consider, for example, the network in 3.4, with nodes *Surfing* and *Fishing* having a common child node *Water pollution* (which we assume can be caused by both fishing and surfing activity). The conditional independence of the parent nodes would not be satisfied, as they would both share a common cause, i.e. *Waves*. To illustrate the recursiveness of the loop, consider the π message from *Surfing* to *Water pollution*. It would convey top-down evidence from *Waves*, which in turn would include evidence from its descendants *Fishing* and *Water pollution*.

Several methods have been developed to deal with the problem of multiply-connected graphs. Exact inference methods all have a complexity that is exponential to the width of the network. Approximate inference methods are designed to reduce the processing complexity, although the trade-off is reduced accuracy of the result. Most approximate inference methods yield message-passing algorithms which can be implemented in a distributed manner, equivalent to the original belief propagation. Note these methods are used not only for networks with loops, but also for networks with other type of complexities, such as high fan-in or a large number of layers.

3.3.5.1 Exact inference methods

- *Clustering/junction tree algorithm*: This method provides exact marginalization of multiply connected Bayesian networks. It entails performing belief propagation on a modified version of the Bayesian network called a *junction tree*. The junction tree is an undirected graph in which groups of nodes are clustered together into single nodes in order to eliminate the cycles. The algorithm can be very computationally expensive, specially for

large-scale networks (Jordan and Weiss 2002).

- *Cutset conditioning*: This method, also called *reasoning by assumption*, also provides the exact marginal probabilities. It involves breaking the loops by finding a small set of variables which, if known (i.e. instantiated), would render the remaining graph singly connected. For each value of these variables, belief propagation obtains the beliefs of the nodes in the the singly connected network. The final value is obtained by averaging the resulting beliefs with the appropriate weights obtained from the normalization constants.

3.3.5.2 Approximate inference methods

- *Loopy belief propagation*: This method implies naively applying the belief propagation algorithm on a network despite it having loops. The formulation would be theoretically incorrect, and the messages would circulate indefinitely through the network due to its recursive nature. Nonetheless, empirical results in error-correcting networks, such as the *turbo code* (Weiss 1997), demonstrate the method provides a good approximation to the correct beliefs. The method has also been applied satisfactorily to other type of network structures, such as the *PYRAMID* network, which resembles those used for image processing (Murphy et al. 1999, Weiss 2000). The resulting beliefs in these networks showed convergence, as opposed to oscillations, after a number of iterations.

- *Sampling/Monte-Carlo algorithms*: These methods rely on the fact that, while it might be infeasible to compute the exact belief distribution, it may be possible to obtain samples from it, or from a closely-related distribution, such that the belief can be approximated averaging over these samples. For large deep networks these methods can be very slow (Hinton et al. 2006).

  The *Gibbs sampling* and *Metropolis-Hastings* algorithm are both special cases of the Markov Chain Monte Carlo algorithm. The first one involves selecting a variable, x_1 for example, and computing a simplified version of its belief based only on the state of its neighbours at time t, such that $Bel(x_1^{t+1}) = P(x_1|x_2^t, ..., x_n^t)$. The process is repeated for all variables using always the latest (most recently updated) value for its neighbours,

e.g. $Bel(x_2^{t+1}) = P(x_2|x_1^{t+1}, x_3^t, ..., x_n^t)$. The second algorithm provides a less computationally demanding alternative, by choosing a value at random for each variable of the distribution and then calculating the *acceptance probability* of the new distribution. Both methods applied to graphical models yield a message-passing algorithm similar to belief propagation.

In *importance sampling* (also called particle filtering), on the other hand, samples are chosen from a similar but simpler distribution than the original joint probability distribution. This simpler distribution can be obtained by simplifying the original graph, for example, by deleting edges. The samples are then re-weighted appropriately.

- *Variational approximation*: Variational methods, such as the *mean field* approximation, convert the probabilistic inference problem into an optimization problem. The basic approach is to choose from a family of approximate distributions by introducing a new parameter for each node, called a variational parameter. These variational parameters are updated iteratively as to minimize the *variational free energy* of the system, which is equivalent to the cross-entropy (Kullback-Leibler divergence) between the approximate and the true probability distributions. When the variational free energy is minimum, the approximate and the true probability distributions are equivalent. More elaborate approximations to the free energy, such as the *Bethe* free energy, provide better approximate marginal probabilities (Jordan and Weiss 2002, Murphy 2001, Winn and Bishop 2005).

  This method has become more popular in recent years due to the high computational cost of sampling methods. It is currently being used by several research groups to model complex systems such as the visual system (Friston and Kiebel 2009, Hinton et al. 2006). Section 3.4.2 describes some of these models.

3.4 Existing models

Bayesian inference has been employed extensively to model different aspects of cortical processing, from single neuron spikes (Deneve 2005) to higher-level functions, such as object perception (Kersten et al. 2004) and decision making (Chater et al. 2006). In this section, the focus

is on models that use Bayesian belief propagation or similar inference algorithms in the context of hierarchical generative models. In particular, Section 3.4.1 describes several implementations of belief propagation using spiking neurons; Section 3.4.2 describes implementations of belief propagation at a higher level of abstraction, specifically those attempting to model object perception in the visual system; and Section 3.4.3 compares different speculative mappings of the algorithm over the cortical laminar circuitry.

3.4.1 Biological models with spiking neurons

There have been several proposals for how spiking neurons can implement belief propagation in graphical models such as Bayesian networks. Three of these models are described in this subsection.

3.4.1.1 Single layer hidden Markov model

The first one, by Rao (2004, 2005, 2006), describes a single-layered recurrent network that is able to perform a simple visual motion detection task. The input to the model is a 1-dimensional 30 pixel image, with a moving pixel. The model contains 30 neurons, each one coding the 30 different states of a hidden Markov model. The states code a specific spatial location (15 locations with 2 pixel intervals), and the direction of motion (leftward or rightward). The firing rate of each neuron encodes the log of the posterior probability (belief) of being in a specific state, such that the neuron with the highest firing rate indicates the state of the world. To model the likelihood function, equivalent to the bottom-up messages, the input image was filtered by a set of feedforward weights (Gaussian functions), which represent the conditional probability function. The prior, or top-down message, was approximated by multiplying the posterior probability at the previous time-step by a set of recurrent weights which represent the transition probabilities between states.

The model was later extended by adding a second layer of Bayesian decision-making neurons that calculated a log-posterior ratio to perform the random-dot motion detection task. A similar implementation using a simple two-level hierarchical network with two interconnected pathways for features and locations, modelling the ventral and dorsal paths, was used to simulate

attention.

The main contribution of this model is that it managed to implement Bayesian inference using equations representing a recurrently connected network of spiking neurons. However, the main limitation of the model is that it does not offer a general solution to implementing belief propagation with spiking neurons, but rather very specific and simple examples with heuristic implementations. The main model consists of just a single layer containing 30 neurons, which does not capture the complexities of belief propagation, nor its many benefits, such as a local and distributed implementation; furthermore, it does not capture the complexities inherent in visual processing. Additionally, the implementation in the log domain requires the use of an approximation to the conditional probability weights, which has not been proven to provide accurate results when the system is scaled up.

3.4.1.2 Liquid state machine model

A more recent model of belief propagation in networks of spiking neurons was provided by Steimer et al. (2009). The model approximates belief propagation in Forney factor graphs, a type of graphical model that is considered more general than Bayesian networks, and therefore can capture all of its properties. The model makes use of liquid state machines composed of *liquid* pools of spiking neurons to represent the function nodes in the factor graph, similar to the conditional probability functions in Bayesian networks. The internal dynamics of each pool of neurons allows it to combine the incoming messages from the corresponding input nodes. Messages from one node to another are transmitted using *readout* populations of neurons which extract the output information from the liquid pools. The readout populations need to be calibrated and trained to map the input synaptic current with desired output message (probability from 0 to 1), encoded using an average population rate. Figure 3.10 shows the neural implementation of belief propagation in a factor graph using the liquid and readout populations of a liquid state machine.

The model was evaluated using two simple examples: a classical inference problem dealing with the transmission of binary information in an unreliable channel, and a more biologically-grounded example dealing with the integration of psychophysical information to elucidate the

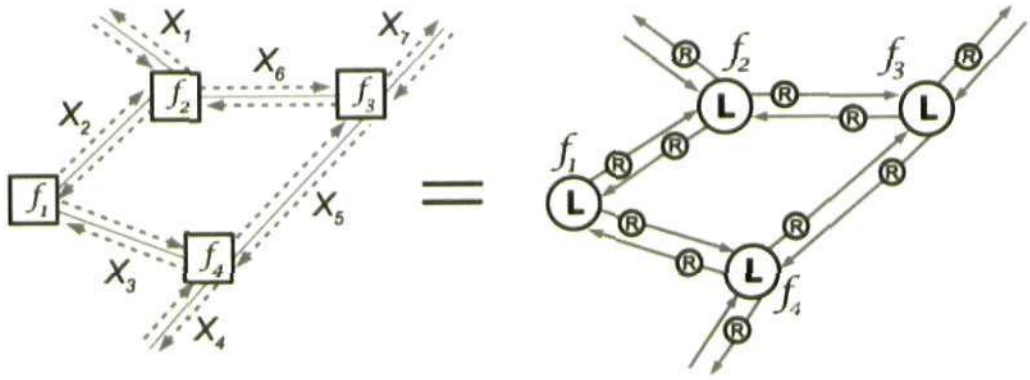


Figure 3.10: Neural implementation of belief propagation in a Forney factor graph using the liquid and readout neuronal populations of a liquid state machine. *Left*) Illustration of a Forney factor graph where the nodes represent factors $f_1, ..., f_4$ (conditional probability functions), and the edges represent variables $X_1, ..., X_7$. Arrows represent the messages exchanged during belief propagation. *Right*) Neural implementation of the Frney factor graph and belief propagation shown in the left. The liquid pools (L) represent the factors of the graph and combine input messages from neighbouring nodes. The messages (and, implicitly, the variables) are encoded by the population rate of readout pools (R) and are injected to the corresponding liquid pools via the synaptic connections (Steimer et al. 2009).

shape and illumination of an object. The population rates of the readout pools, resulting from the network dynamics, were in agreement with the direct numerical evaluation of belief propagation.

Although both networks consisted of a very small number of binary variables (9 and 4 respectively), the authors claim the model can be generalized to more large-scale and complex scenarios. Nonetheless, the number of neurons required to do this, both for the liquid and readout populations, would be extremely high and thus very expensive from the computational perspective. According to the authors, a current line of research aims at increasing the coding efficiency of the neuron pools by making use of a place-coding scheme. A further limitation of scaling up is related to the accuracy of the results in networks with several hierarchical levels. It was shown that messages deep in the network were less correlated to the exact numerical values, than those near the input layer.

3.4.1.3 Local inference circuits model

Concurrent to the publication of the previous model, Litvak and Ullman (2009) published an alternative implementation of belief propagation using spiking neurons. The latter employs populations of standard leaky integrate-and-fire neurons to implement the belief revision algorithm in pairwise Markov random fields. Belief revision is analogous to belief propagation except that it replaces the *sum* operation with the *max* operation, i.e. uses the max-product instead of the sum-product algorithm, thus obtaining the maximum-a-posteriori estimate (also called the most probable explanation) instead of the posterior marginal probability. Additionally, the algorithm is implemented in the log domain, which leads to a final neuronal implementation based on a max-sum scheme, called *belief consolidation*. Pairwise Markov random fields are a type of undirected graphical model, which share many properties with directed graphical models (Bayesian networks), but are not interchangeable (see Section 3.3.3).

To implement the belief consolidation algorithm, the model uses building blocks called local inference circuits (LINCs). Each neuronal LINC is connected to other LINCs according to the graph structure, and propagates the same message to all neighbours. Each LINC roughly implements the operations performed locally by each node in the graph using smaller elementary circuits that approximate the two mathematical operations: a linear summation circuit and a maximization circuit. The model uses populations of leaky integrate-and-fire neurons to implement these computations. The synaptic weights between the different elementary circuits define their specific functional properties. The mean rate of the neural populations during short periods of time (few tens of milliseconds), represents the values of messages computed during the inference process.

Each neuronal LINC uses N (number of neighbours) x S (number of states) weighted maximization circuits, which compute the maximum value for each state of the input nodes. Before finding the maximum value, the circuit uses a linear summation element to add the corresponding weight to each input message (in the log domain, weights are additive). The weighted maximum results for each state are then combined in the N corresponding summation circuits. The vector of single valued outputs of each summation circuit represents the output message of

the node encoded by the LINC. In a final step, the output message is normalized via a normalization circuit where the different values are recurrently inhibiting each other. This is achieved by connecting the excitatory population of all summation units to a central inhibitory population. A schematic representation of a neuronal LINC node is shown in Figure 3.11.

The model was tested using two sets of graphs, one with 12 hidden binary variables and one with 6 hidden ternary variables. In each case 100 different random configurations of the node weights and evidence values were tested and compared with the original analytical methods. Results showed the neuronal circuit was able to effectively approximate the marginal distributions, although the accuracy decreased when using ternary variables as compared to binary variables. The inaccuracy of the model was shown to arise not only from the sub-circuit's approximations (sum, max and normalization), but from inherent network phenomena such as the evolving desynchronization in subpopulations.

With regard to the scalability, the model can map any arbitrary graph structure and discrete variables with any number of states, with a linear relation between the number of neurons and the number of nodes. However, for large-scale networks and variables with many states, the number of neurons might be prohibitive (6 hidden variables each with 3 states require over 16,000 neurons). The speed of the computation provides a biologically realistic inference time ($\approx$ 400 ms) due to the highly distributed implementation. The model also attempts to map the different algorithm operations onto the cortical laminar circuits, as described in Section 3.4.3.

A comparison between the most significant features of the previous two models is depicted in Table 3.1, including a summary of the main advantages and drawbacks of each model.

3.4.1.4 Electronic implementation of networks of spiking neurons

An emerging and rapidly growing field of research is dedicated to the implementation of realistic spiking neural circuits in hybrid analog/digital very large scale integration (VLSI) devices. Recent advances have allowed the implementation of winner-take-all networks in the VLSI devices, which has lead to the development of simple state-dependent systems (Neftci et al. 2010). Simple graphical models, such as factor graphs and belief propagation, can be approximated using winner-take-all networks with state-dependent processing. Examples of graphical models,

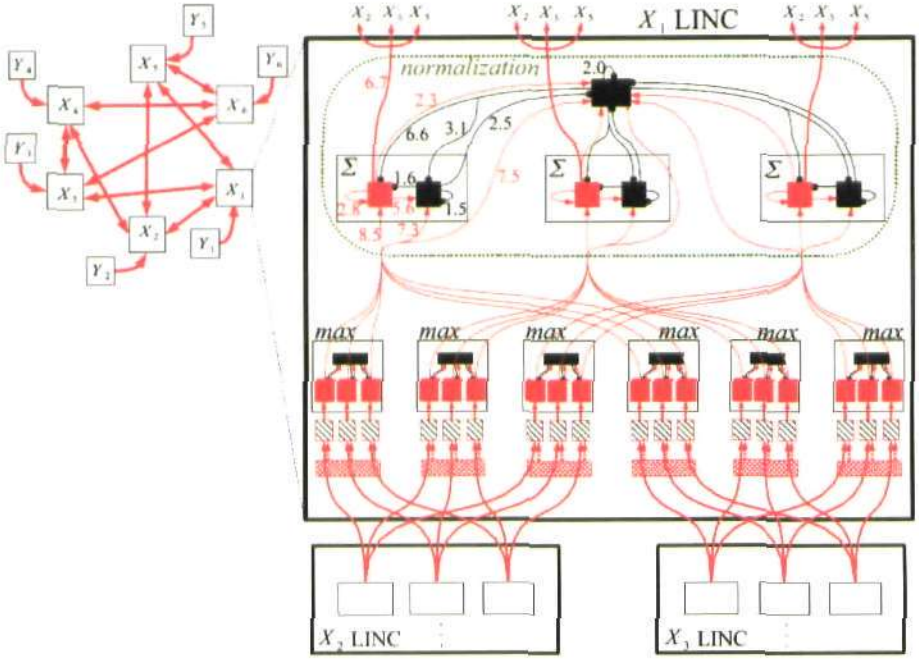


Figure 3.11: Neuronal local inference circuit (LINC) implementing the operations of a node in a Forney factor graph with 6 input nodes (Y_i) and 6 hidden variables (X_i). Each LINC is built from populations of leaky integrate-and-fire neurons (small red, black and dashed rectangles), which implement the two basic operations: weighted maximization circuits (*max* frames) and summation circuits (Σ frames). The main black frame shows the neuronal LINC for the variable X_1, which receives input from neighbour nodes and X_5 (although due to size limitations only projections from nodes X_2 and X_3 are shown). For each neighbour node, and for each of the three states of X_1, a maximization node finds the maximum of the weighted message values. Note a linear summation circuit adds the corresponding weight (log domain) to each input message prior to the maximization step. The set of weighted maximum results for each state are then combined in the three corresponding summation circuits. The three sums are then normalized by a normalization circuit (green dotted frame), which contains a recurrently connected inhibitory population (black rectangle in the centre). The vector of single valued outputs of each summation circuit represents the output message of X_1, which will be propagated to all of its neighbours (Litvak and Ullman 2009).

	Steimer, Maas and Douglas	**Litvak and Ullman**
Type of graph	Forney factor graph	Pairwise Markov random field
Algorithm	Belief propagation	Belief consolidation ≡ log belief revision
Neuron model	*Leaky integrate-and-fire*	*Leaky integrate-and-fire*
Building blocks	Liquid state machines (liquid and readout pools)	Local inference circuits (LINCs) (summation and weighted maximization circuits)
Number of neurons	$\approx 1592/state/node$ [a]	$\approx 744/state/node$ [b]
Coding strategy for messages	Population rate of trained readout neurons	Population rate of output summation circuits in each LINC
Coding strategy for conditional probability functions	Neural dynamics of liquid state machine	Additive weight (log domain) in weighted maximization circuit
Use of inhibitory connections	Dynamics of liquid pool	Winner-take-all implementation in maximization circuits and normalization in summation circuits
Advantages	Scalability, generality and relatively simple structure (two types of neural populations liquid and readout pools)	Scalability, generality, realistic biological time scale, mapping onto cortical circuitry.
Drawbacks	Number of neurons, loss of accuracy in deeper layers, tuning of liquid pool dynamics to factors and training of readout populations	Number of neurons, loss of accuracy in deeper layers, restricted to log domain and belief revision algorithm.

[a] Between 300 and 1950 neurons per factor node (liquid pool) + 343 neurons per binary message (readout population) x 4 neighbour nodes (equivalent to Litvak and Ullman, 2009) = 1125 (average) + 2058 ≡ 1592 neurons per state per node

[b] 12 hidden binary variables = 13,512 ; 6 hidden ternary variables = 16,656 neurons; average = 744 neurons per state per node

Table 3.1: Comparison of two implementations of graphical models and belief propagation using spiking neurons.

using two of the implementation methods described in this section (Litvak and Ullman 2009, Steimer et al. 2009), have already been implemented using the state-dependent VLSI technology (Emre Neftci, personal communication).

Although the technology is still at a very early stage and the scalability of the VLSI spiking neural networks is limited, it provides a starting point for the development neuromorphic hardware capable of reproducing graphical models with cortical functionality.

3.4.2 Functional models of visual processing

This subsection focuses on models based on generative modelling approaches, which employ Bayesian networks/belief propagation or similar implementation methods. Specifically, we describe models which deal with visual perception (recognition, reconstruction, etc.) and have biologically grounded architectures. The literature in this area is very extensive so only models most relevant to this thesis are included. To facilitate comparison between models, they have been grouped according to the inference method employed (exact inference, sampling approximation, or variational approximation), although the classification is not strict as some models share characteristics of several methods. A summary and comparison of the models is included at the end of this subsection.

3.4.2.1 Models based on exact inference methods

The model proposed by Epshtein et al. (2008) extends a well-known feedforward object recognition model, namely Ullman's fragment-based hierarchical model described in Section 2.1.2. A hierarchy of informative fragments and its corresponding smaller sub-fragments are learned for each class of objects. This information is stored using a factor graph where each variable represents an object fragment, which can take N different values/states indicating the position of that fragment within the image (a value of 0 indicates the fragment is not present). The relation (conditional probability function) between a sub-fragment and its parent fragments depends on the coordinate difference between the locations of child and parent fragments, and not on their absolute position. This allows the model to perform recognition with certain position invariance.

The model computes the similarity between each low-level feature and the image at N different locations, which is used as input evidence for the network (similar to dummy nodes in Bayesian networks). A simple bottom-up sweep of the belief propagation algorithm then obtains the probability distribution for each variable (i.e. presence/location of each fragment), including that of the root node, which represents the class. Note, unlike conventional feedforward methods, the model computes the relative likelihoods of all class sub-hierarchies given the stimuli (i.e. there is a graph for each class of objects), leading to multiple alternatives at each level of the model. Later, a top-down cycle obtains the optimal value for all the object parts given the state/location of the root/class node, correcting most of the errors made during the bottom-up pass. This provides not only object recognition, but a detailed interpretation of the image at different scales and levels of detail. The model was tested a large number of natural images belonging to three different object classes.

Unlike most related models (Riesenhuber and Poggio 1999, George and Hawkins 2009, Murray and Kreutz-Delgado 2007, Lewicki and Sejnowski 1997), where nodes represent locations and states represent features, the model by Epshtein et al. (2008) uses nodes to represent features and states to represent locations. In essence, the network includes a fixed hierarchical representation of all the possible combinations of features and subfeatures of a class of objects. The graph is a singly connected tree (no loops and a single parent per node) which makes tractable the use of belief propagation to perform exact inference.

However, the previous properties imply that features are not shared within the same object (each feature can only be present at one given location), amongst different objects of the same class (the graph for each class is singly connected), or within objects of different classes (there is an independent network for each object class). This lack of overlap between features speaks for an inefficient coding strategy, as low-level features of distinct objects are likely to be similar. Additionally, the model is restricted to a set of informative learned fragments, which, for example, limit its ability to explain retinotopic contour completion at an arbitrary (less informative) object region.

A second model falling into this category is that proposed by Chikkerur et al. (2010). It uses

the output of the standard HMAX model (Serre et al. 2007b), described in Section 2.1.2, as the input to a Bayesian network which simulates the effects of spatial and feature-based attention (modelling the prefrontal cortex and the lateral intraparietal regions). The network consists of a node L, encoding the location and scale of the target object; a node O, encoding the identity of the object; and a set of nodes X_i that code the different features and their locations. The feature nodes receive evidence from the HMAX-based preprocessing network, which extracts a set of high-level features (roughly corresponding to V2/V4 receptive fields) from the image. At the same time, they receive top-down feedback from the object location (L) and identity (O), using conditional probability distribution $P(X_i|O,L)$. This distribution is constructed based on whether the object contains a given feature (obtained from the HMAX parameters), and whether the feature location matches the spatial attention location (Gaussian centred around that location).

The model is successful at capturing several attentional effects such as the pop-out effect and feature-based and spatial attention, and predicts eye fixations during free viewing and visual search tasks. However, it cannot be considered a generative model of the visual system as it cannot produce input images, i.e. the model relies on the HMAX framework to analyze and extract features. This means the effects of attention on lower visual areas cannot be modelled. The Bayesian network is limited to a relatively abstract implementation of the high-level interactions between the ventral and dorsal pathways. Exact inference can be performed using a single up and down pass of the belief propagation algorithm due to the simplicity of the network, where only the feature layer has more than one node.

Another interesting architecture, and one which takes into account temporal as well as spatial information, is the Hierarchical Temporal Memory (HTM) proposed by George and Hawkins (2009). The model assumes that images are generated by a hierarchy of causes, and that a particular cause at one level unfolds into a sequence of causes at a lower level. An HTM can be considered a special type of Bayesian network which contains a variable coding the spatial patterns, and a second variable coding sequences of those spatial patterns (represented using a Markov chain).

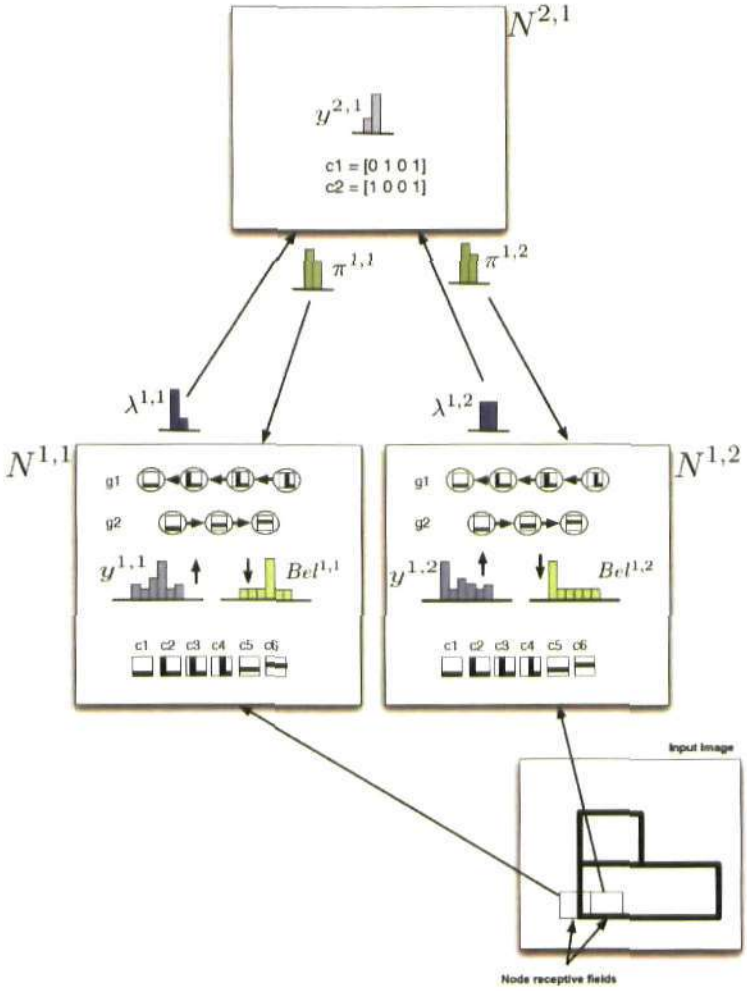


Figure 3.12: Toy example of belief propagation in Hierarchical Temporal Networks (HTM). The network segment shown includes two nodes at level 1 ($N^{1,1}$ and $N^{1,2}$) and one node at level 2. Each node at level one has six coincidence patterns (spatial variable) and two Markov chains (temporal sequence variables), which are illustrated qualitatively to correspond to visual patterns within the receptive field of each level 1 node. The Markov chain *g1* corresponds to a corner moving left and Markov chain *g2* corresponds to a horizontal line moving upward. The distribution $y^{1,1}$ represents the bottom-up likelihood of coincidence patterns in node $N^{1,1}$ given the evidence. The bottom-up message $\lambda^{1,1}$ represents the bottom-up likelihood of Markov chains in node $N^{1,1}$ given the evidence. Note $\lambda^{1,2}$ shows a flat distribution reflecting the fact that the horizontal line pattern participates in both Markov chains. The parent node $N^{2,1}$ has only two coincidence patterns in this toy world, corresponding to a concatenation of the lower level bottom-up messages. The coincidence pattern likelihood $y^{2,1}$ indicates that pattern 2 is more likely given the input messages. Even though child $N^{1,2}$ receives bottom-up ambiguous information about its Markov chains, integrating more global information gives rise to a peaked top-down distribution, $\pi^{1,2}$. Note the distributions shown are just qualitative examples and do not correspond to any real computation. From George and Hawkins (2009).

During the learning stage, HTMs attempt to discover the causes underlying the sensory data. Each node makes use of a spatial pooler that learns the most common input spatial patterns, and a temporal pooler that groups these patterns according to their temporal proximity and assigns them a label. For example a set of corner lines at different positions (input spatial patterns), could be grouped into a common temporal group labeled *corner*. Note the terms *temporal group*, *sequence* and *Markov chain* represent the same concept in an HTM network.

The spatial pooler in the parent node combines the output of several lower-level nodes, which takes the form of a probability distribution over the temporal groups of those nodes. This allows it to find the most common co-occurring temporal groups below, which then become the alphabet of spatial patterns in the parent node, e.g. features of a face (eyes, nose, mouth) which always move together. The concept is similar to that of invariant features obtained by the complex layers of the HMAX model (Serre et al. 2007c).

The learning process is repeated throughout the hierarchy to obtain the causes at the highest level. As a result, a tree structured Bayesian-like network is obtained, based on the spatio-temporal characteristics of the inputs which reflect that high-level features vary more slowly than low-level features. This strategy is similar to that employed by the trace rule in the Visnet model (Deco and Rolls 2004) and slow-feature analysis (Wiskott and Sejnowski 2002) (see Section 2.1.2).

During the inference stage, a variant of the belief propagation algorithm adapted to HTM networks propagates sensory evidence from multiple low-level regions (conveying competing hypotheses), which converge on a high-level cause, leading to recognition. Top-down feedback then proceeds, analogously to Bayesian networks, disambiguating lower level patterns. The process is illustrated in Figure 3.12, using a toy example with three HTM nodes in a two-level hierarchy.

The algorithm is different from the original belief propagation in that it takes into account the temporal information included in the Markov chains of each node to compute the belief and output messages. In standard Bayesian networks (Pearl 1988) each node represents a single random variable. In addition, to solve the problem of nodes with multiple parents, the author

proposes a trivial extension of the algorithm using the Noisy-OR gate. However, this method is only valid for graded variables (see Section 3.3.4 and Pearl (1988)), which is not the case for variables in HTM networks. Similarly, the problem of networks with loops is solved by implementing loopy belief propagation, which is claimed to provide good results, although no evidence is provided.

Model simulations shows succesful recognition (72%) of 48 line drawing objects (32 x 32 pixels) despite translations, distortions and clutter. When tested on the standard *Caltech-101* benchmark of natural images, the performance decreased significantly (56%); although when using their own 4-category testset of natural images, the accuracy was very high (92%). Preliminary results also suggest top-down feedback in the model can account for segmentation, feature binding, attention and contour completion. Only the last phenomenon is explicitly demonstrated, by firstly recognizing a Kanizsa square (input image) as a square (high-level cause), and later alowing top-down feedback to increase the response of nodes coding the retinotopic location of the illusory contours. Due to the significant similarities between HTMs and the model proposed in this thesis, a more detailed comparison between them is included in Section A.

3.4.2.2 Models based on sampling approximation methods

The first model in this subsection was described in a landmark paper by Lee and Mumford (2003). From a relatively abstract perspective, the belief propagation approach was proposed to account for processing in the ventral visual pathway (V1, V2, V4 and IT). The visual cortex was suggested to represent beliefs or conditional probability distributions on feature values, which are passed forward and backward between the areas to update each other's distribution. The authors extended this model, proposing an alternative way of implementing approximate Bayesian inference by using a sampling method called particle filtering. This mathematical tool approximates high-dimensional probability distributions using a set of sample points or particles and an attached set of weights that represent their probabilities. The essential idea is to compute for each area not only one hypothesis for the true value of its set of features, but a moderate number of hypotheses. This allows multiple high-probability values to stay alive until a larger number of feedback loops have had a chance to exert an influence. However, no

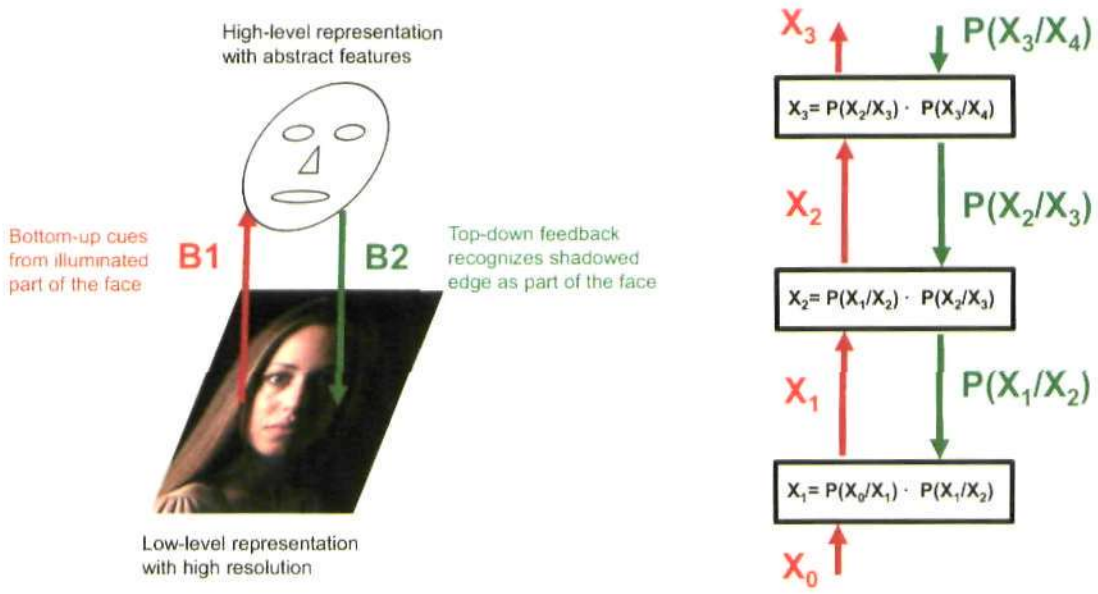


Figure 3.13: Bayesian belief propagation architecture applied to the visual system. a) Initially, bottom-up cues from the illuminated part of the face (B1) cause a *face* hypothesis to become activated at the higher levels. Then information about the likely features and proportions of a face is conveyed through top-down feedback (B2) to the lower-level high resolution buffer. Re-examination of the data results in a reinterpretation of the faint edge in the shadowed area as an important part of the face contour. b) Each area computes a set of beliefs, X_i, based on bottom-up sensory data (X_{i-1}) and top-down priors ($P(X_i/X_{i+1})$, which are integrated according to the Bayesian inference equation. Beliefs are continually updated according to changes in earlier and higher areas to obtain the most probable distribution of causes at each level. Adapted from Lee and Mumford (2003).

practical implementation of this theoretical approach was provided by the authors.

Nonetheless, this theoretical paper has strongly inspired and motivated the present thesis, and provides an intuitive example which allows one to better understand the concept of how belief propagation can be applied to visual processing. Consider the shadowed face example shown in Figure 3.13. Initially, bottom-up cues from the illuminated part of the face cause a *face* hypothesis to become activated at the higher levels. Then information about the likely features and proportions of a face is conveyed through top-down feedback to the lower-level high resolution buffer. Re-examination of the data results in a reinterpretation of the faint edge in the shadowed area as an important part of the face contour. This new detailed information can then be used

by the higher levels to infer additional characteristics of the image, such as the precise identity of the face.

Lewicki and Sejnowski (1997) demonstrate the efficiency of Gibbs sampling in learning higher level parameters in Bayesian networks. A simple 3-level network of stochastic binary variables with a 5x5 pixel input image is used to discover higher level motion patterns from the input image correlations (the *Shifter* problem). Importantly, feedback from the third layer, containing the global direction of motion, is used to disambiguate the local shift direction in layer two. The combination of information from multiple parents/causes was approximated using the Noisy-OR gate, previously described in Section 3.3.4.

Hinton et al. (2006) proposed a new type of network called a *deep belief net* which is composed of a Bayesian network (directed acyclic graph) with two undirected associative memory layers at the top. The motivation for this model is to ease the intractable unsupervised learning process in hierarchical Bayesian networks, where, in order to learn the weights of the bottom layer it is necessary to calculate the posterior probability which depends not only on the likelihood (bottom-up data) but also on the prior (top-down data). In other words, as a result of the explaining away effect, the weights of all the higher layers are required. Further, it is necessary to sum over all possible configurations of the higher variables in order to obtain the bottom layer *prior.*

The authors introduce the concept of *complementary priors*, which are prior distributions that, when multiplied by the corresponding likelihood function, yield a posterior distribution which can be factorized. This implies eliminating the explaining-away effect, thus making each hidden layer independent of its parents' weights. This yields a network which is equivalent to a Restricted Boltzmann Machine, i.e. a network with an independent hidden layer of binary variables with undirected symmetric connections to a layer of observed nodes. Under these conditions a fast learning algorithm is derived which obtains the approximate parameters of the network layer by layer. First, a visible layer (input image) is used to train the bottom hidden layer of the network. After learning the weights of the hidden layer, the activations of that layer, given the input image, are used as the input data for the hidden layer above, thus always

maintaining the 2-layer structure characteristic of Restricted Boltzmann Machines.

The above learning method can be seen as a variational approximation wherein the constraint is that the weights in the higher levels ensure the *complementary priors* condition, therefore yielding a factorial posterior distribution. However, as weights in higher-levels are learned, the priors for lower layers cease to be complementary, so the weights used during inference are incorrect. Nonetheless, it can be shown that each time the weights of a layer are adapted, the variational lower bound on the log probability of the training data is improved, consequently improving the overall generative model. The weights of the model are then finely tuned in a final stage by performing an up and down pass of a variant of the *wake-sleep* algorithm (Hinton et al. 1995). Although the learning is unsupervised in the directed layers, the top two associative layers can be used to learn labeled data.

Inference is achieved by a single up pass along the bottom directed layers, yielding the binary states of the units in the lower associative memory layer. Further Gibbs sampling or free-energy optimization activates the correct label unit at the top layer. The performance of the model on the MNIST digit recognition task was superior to that of previous models, including Support Vector Machines and back-propagation. This demonstrates that generative models can learn many more parameters than discriminative models without overfitting. The model is still limited in that top-down feedback during inference is restricted to the top associative layers. Additionally, it does not deal systematically with perceptual invariances. Instead, invariance arises as a consequence of the wide range of sample images that can be generated by the model for each given category.

3.4.2.3 Models based on variational approximation methods

The free-energy model proposed by Friston (Friston 2003, 2005, Friston et al. 2006, Friston and Stephan 2007, Friston and Kiebel 2009, Friston 2010) has already been described in some detail in Section 3.1.3. It is based on a variational approximation and therefore converts the complex inference problem into an optimization task which tries to minimize the free-energy between the true posterior distribution and and the recognition distribution. By assuming a Gaussian approximation (Laplace assumption) to the recognition distribution, optimization becomes equiv-

alent to finding the means of the unknown causes of sensory data given the generative model. The specific form of the generative model is given by the equations of a hierarchical dynamic model which impose structural and dynamical constraints on the inference process. Solving these equations implies implementing a message-passing algorithm reminiscent of the predictive coding scheme.

Friston (2005) then reviews anatomical and physiological data from the brain, suggesting the proposed hierarchical dynamical system and message-passing scheme could be implemented by the cortex. At the same time brain responses related to perception and action can be understood in terms of the proposed model. However, the model remains in a relatively theoretical form and is only applied practically to two simple scenarios: a birdsong recognition problem, and a 4-pixel image recognition. The second example, more relevant for this section, comprises a a 2-layer network which illustrates the dynamics of the free-energy model and how the prediction error is reduced after the parameters are gradually learned.

A similar approach was previously implemented by Rao and Ballard (1999) using the Kalman filter, which is derived from the Minimum Description Length principle, similar in flavour to free-energy minimization. The model could have been included in this section as it employs a variational approximation, but was previously described in Section 2.2.3, together with other *predictive coding* models of the visual system.

The model by Murray and Kreutz-Delgado (2007) also attempts to solve several visual perceptual tasks such as recognition or reconstruction, formulating them as inference problems in a stochastic generative model. The joint probability distribution is defined using the *neighbouring layer conditional probability* (NLCP), which states that the nodes of a layer only depend on the nodes of its immediate neighbouring layers (closely related to belief propagation in Bayesian networks). The NLCPs can conveniently be formulated using Boltzmann-like distributions. A variational approximation (factorial Bernoulli distribution) is employed to deal with the intractable exact inference problem. This leads to the development of a simplified generative model which can be implemented using a hierarchical dynamic network with feedforward, feedback and lateral connections. The model places a strong focus on overcomplete sparse

representations (suggested by experimental evidence), which are enforced during the learning stage, and improve recognition performance.

A four-layer network with a 64x64 pixel input image simulates object recognition in the visual system. The network managed to correctly recognize, segment and reconstruct occluded versions of the trained images, although no invariance to position and size transformations is achieved. The study illustrates some interesting properties, such as the possibility of simulating *imagination* by running the network generatively (i.e. top-down and not bottom-up input); and expectation-driven segmentation, whereby the top-down input (e.g. prior expectations) improves recognition in cluttered scenes. However, the model fails to provide mechanisms for position and scale invariance during recognition. Furthermore, despite being based on a generative model, the resulting dynamic network derived from the simplified model is far from the original belief propagation scheme.

3.4.2.4 Comparison and conclusions

This subsection has outlined some of the attempts to model visual perception in the brain using the generative modelling approach, and in particular those employing algorithms similar to belief propagation. Table 3.2 lists the models, comparing the type of network, inference algorithm and results obtained in each case.

The complexity that emerges from the large-scale and intricate cortical connectivity means exact inference methods are intractable, making it necessary to use approximate solutions such as loopy belief propagation (George and Hawkins 2009), sampling methods (Hinton et al. 2006, Lee and Mumford 2003, Lewicki and Sejnowski 1997) or variational methods (Murray and Kreutz-Delgado 2007, Rao and Ballard 1999, Friston 2010). Sampling methods typically maintain the probabilistic nature and structure of Bayesian networks, while variational approximation methods yield a hierarchical dynamic network which deals with the optimization problem (minimizing the difference between the approximate and the true posterior distributions). Nevertheless, in both cases the resulting dynamics lead to local, recursive message-passing schemes reminiscent of belief propagation.

Exact inference is only possible when the generative model avoids physiological constraints,

Model	Type of network	Inference algorithm	Results
Epshtein & Ullman, 2008 (Fragment-based model)	Factor graph, singly-connected, one graph per object class.	Belief propagation	Natural images (120x210 px), 3 object classes, recognition, position invariance, feedback corrects information on object fragments.
Chikkerur et al., 2009 (Attention model)	Bayesian network, 4 layers (only one layer with more than one node)	Belief propagation	Models attentional effects, not image recognition/reconstruction. Images first processed with HMAX model (not Bayesian).
George & Hawkins, 2010 (Hierachical Temporal Memory)	HTM network - Bayesian network with Markov chains inside each node	Belief propagation adapted to HTMs (+ loopy)	Line-drawing images (32x32 px) and natural images (160x160 px), recognition, reconstruction, contour completion of Kanizsa square.
Lee & Mumford, 2003	Bayesian network	Belief propagation with sampling (particle filtering)	Theoretical
Hinton et al. 2006 (Deep belief nets)	Bayesian network (3 layer DAG), 2 top undirected associative memory layers	Gibbs sampling with variational approximation (complementary priors)	Hand-written number images (28x28 binary px), recognition (better than previous methods), implicit invariance due to generative variability.
Lewicki & Sejnowski, 1997	Bayesian network (2 layers)	Belief propagation with Gibbs sampling	Line-drawing images (5x5 px), learns higher-order correlations, feedback disambiguates lower layers.
Friston, 2010 (Free-energy model)	Hierarchical dynamic network (example uses simple 3 layers)	Message-passing derived from variational approximation, predictive coding	Mainly theoretical; 1D input image (4 px), reduction of prediction error.
Rao & Ballard, 1997 (Predictive coding)	Hierarchical dynamic network (examples uses 2 layers)	Message-passing, Kalman filter	Natural object images (128x128 px), recognition, occlusion and rotation invariance, feedback reconstruction and RF learning.
Murray & Kreutz-Delgado, 2007	Hierarchical dynamic network with lateral connections, 4-layers	Message-passing derived from variational approximation	Natural object images (64x64 px), recognition, occlusion invariance, image reconstruction.

Table 3.2: Comparison between models of visual processing based on generative modelling approaches, similar to Bayesian networks and belief propagation.

such as multiply-connected networks, and a shared dictionary of low-level features (Epshtein et al. 2008); or models exclusively higher level phenomena such as attention, relying on non-Bayesian object recognition models (Chikkerur et al. 2009).

The results of model simulations on real-world data are still limited. Some models remain purely theoretical (Lee and Mumford 2003), or provide simple toy examples (Friston and Kiebel 2009, Lewicki and Sejnowski 1997, Hinton et al. 2006). Those that use bigger and more complex input images fail to account for certain aspects of object perception, such as position and scale invariance (Rao and Ballard 1997, Murray and Kreutz-Delgado 2007, Chikkerur et al. 2009), or feedback reconstruction (e.g. illusory contour completion) (Hinton et al. 2006, Epshtein et al. 2008); or are not implementing rigorous, theoretically-grounded generative models (George and Hawkins 2009).

Generative models have been described as the *next generation of neural networks* (Hinton et al. 2006). However, their application to visual perception using realistic data is still at a very early stage. Much work needs to be done exploring the different approximate inference methods, network structures, learning methods and scalability of these networks, which allow them to deal with natural image statistics and capture the wide variety of perceptual phenomena, while using realistic physiological parameters.

3.4.3 Cortical mapping of models

The homogeneous, local and distributed implementation of belief propagation in graphical models is reminiscent of the concept of a canonical local circuit that has been suggested to exist in the mammalian cortex. These ubiquitious circuits, shared by many species and cortical areas, are repeated within cortical columns of a few hundred microns, which contains neurons with similar feature tuning properties. Several studies have focused on a theoretically precise mapping between the local structures of graphical models and the layered cortical structure within a cortical column. These also describe the intercortical projections which lead to the larger scale functionality.

Two of the cornerstone studies that have set the theoretical grounds for understanding cortical computation within the hierarchical Bayesian inference framework (Lee and Mumford 2003,

Friston et al. 2006) have sketched the role that some of the laminar connections may play. They both suggest the bottom-up messages of belief propagation may be encoded in the activity of pyramidal neurons in the superficial layers 2/3; while top-down messages might result from activity in deep layer 5 pyramidal neurons.

Litvak and Ullman (2009) provide a more precise and comprehensive account of how the anatomical and physiological aspects of the cortical local circuitry can be mapped onto the elements of graphical models, more precisely those implementing their belief consolidation (max-sum operations) model. Their study provides evidence for the existence of local functional subnetworks which may represent the states of variables, and a higher-level organization (possibly cortical columns) which groups several possible states into variables. According to the authors, empirical data suggests these subnetworks or neuronal cliques are characterized by having excitatory pyramidal neurons and inhibitory fast-spiking basket cells, which are strongly interconnected, and receive input from a common source.

The maximization nodes in the model (see Section 3.4.2) are hypothesized to be implemented in superficial cortical layers by independent minicolumns, small ensembles of neurons organized in vertical arrays, covering approximately 50 microns. Each minicolumn computes the maximum of several weighted input messages, making use of several neural subnetworks with *central inhibition. Double-bouquet inhibitory cells reciprocally connected to all inputs drive* the nonlinear responses. Feedforward projections from the superficial neuronal subnetworks of these minicolumns in a lower cortical area terminate on a neuronal subnetwork in layer 4 of a higher cortical area. The dynamics between excitatory and inhibitory neurons in the target subnetwork allow it to produce a linear response to the sum of its inputs, providing the corresponding cortical mapping to the linear summation circuits proposed in the model. Further details and a thorough review of evidence in support of the proposed functional roles for cortical microcircuits are included in Litvak and Ullman (2009).

George and Hawkins (2009) also provide a detailed description of the possible mapping between cortical microcircuits and their belief propagation model. The mapping is based on their specific formulation of belief propagation adapted to the Hierarchical Temporal Memory networks (see

Section 3.4.2). Similar to the previous mapping, cortical columns implement the different nodes or variables (e.g. coding for a specific region of the input image), while minicolumns represent the different possible states or features at that location (e.g. different orientations).

According to the authors, projections from lower cortical to layer 4 levels are responsible for the storage and detection of coincidence patterns. The synaptic connections between these layers represent the co-ocurrence of patterns on its inputs. Layer 4 then projects onto layer 2/3 pyramidal cells which are assumed to behave as complex cells which respond to invariant features or motion sequences. Thus, layer 2/3 is responsible for the calculation of the feedforward Markov chains' (groups or sequences) states as suggested by the high density of lateral connections. At the same time, anatomical connections suggest these neurons project the Markov chain information to higher cortical levels, and incorporate high-level information, received via layer 1 projections, into the computation of the Markov chains. Layer 5 pyramidal neurons with dendrites in layers 1, 3 and 4 are responsible for the Belief calculation. Finally, layer 6 neurons with dendrites in layer 5 compute the feedback messages for lower regions.

Further inspection of the proposed mappings reveals several key similarities and differences between them, which are summarized in Figure 3.14. Each variable or graph node is roughly understood as a cortical functional column, containing smaller functional units or minicolumns, which correspond to the different variable states (George and Hawkins 2009, Litvak and Ullman 2009). Feedforward outgoing messages from a node are assumed to originate from pyramidal cells in layer 2/3 (George and Hawkins 2009, Litvak and Ullman 2009, Lee and Mumford 2003, Friston et al. 2006). Feedback outgoing messages originate from pyramidal cells in the infragranular layers (Friston et al. 2006), either layer 5 (Lee and Mumford 2003) or layer 6 (George and Hawkins 2009). Feedforward incoming messages from lower cortical areas target layer 4 neurons (Friston et al. 2006, Litvak and Ullman 2009, George and Hawkins 2009).

However, there are two different neural populations that could potentially encode the incoming feedback messages from higher-levels: neurons in supragranular layers (Litvak and Ullman 2009, Friston 2010), more precisely, in layer 1 according to George and Hawkins (2009); or neurons in infragranular layer 6 (Litvak and Ullman 2009, Friston et al. 2006). Both of them

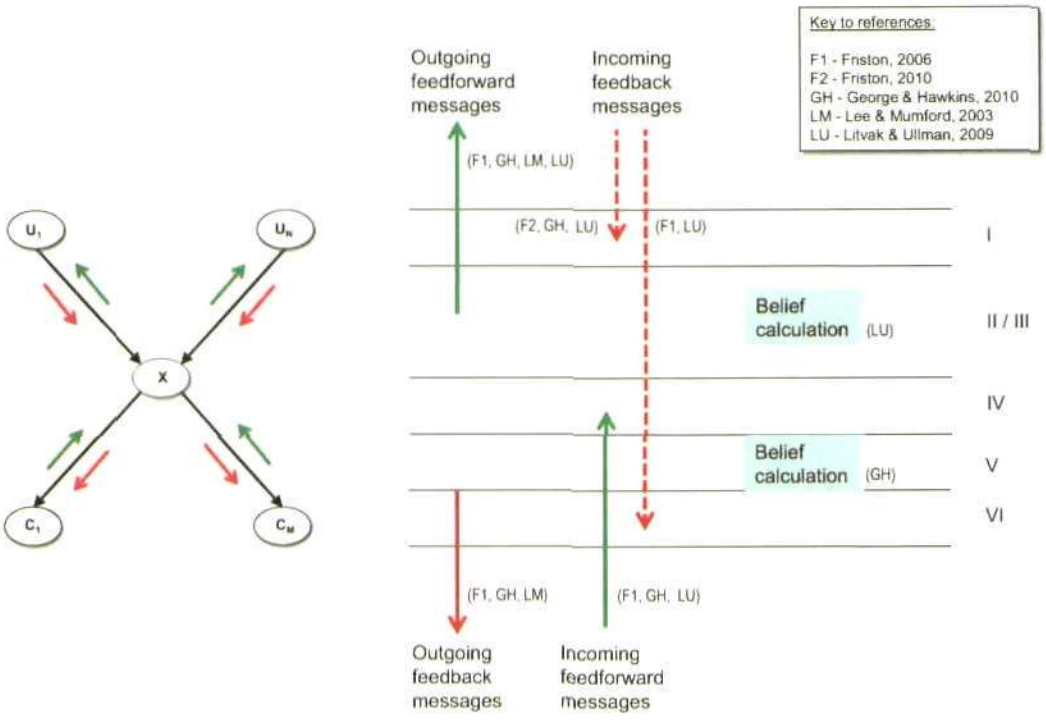


Figure 3.14: Schematic comparison between several proposed mappings between belief propagation and the cortical layers. *Left*) Local message-passing scheme dictated by belief propagation in a Bayesian network. *Right*) Potential mapping of this scheme on the laminar cortical structure according to four different key references which are labelled in the top-right box. Arrows represent the origin or target layer of the four different incoming/outgoing feedforward/feedback message types, together with the relevant references which support this view. Additionally, the diagram also compares the cortical layer hypothesized to implement the belief calculation in two of the proposed mappings.

are major target of feedback connections. The mapping for the calculation of the belief (convergence of feedforward and feedback information) is also a major point of disagreement. While George and Hawkins (2009) suggests it occurs at layer 5, (Litvak and Ullman 2009) suggests either the synapses between layer 6 and 4 neurons, or the strongly laterally connected layer 2/3 neurons, are responsible for the higher and lower information interaction. The discrepancy might result from the ambiguous definition of *belief*, which can be considered equivalent to the output messages in some of the specific algorithms (Friston 2010, Litvak and Ullman 2009, Lee and Mumford 2003).

The mapping of belief propagation models to the cortical architecture is highly speculative and

an over generalization and a simplification of the function of the different cortical layers. The mapping is based on incomplete anatomical and physiological data regarding the standard six-layered sensory cortex structure. The different belief propagation operations are mapped to specific cortical layers, ignoring the fact that these operations may be the result of complex interactions involving several layers simultaneously. Furthermore, the proposed mappings ignore vast amounts of known details, such as the different neuronal types and existing connections (Thomson and Lamy 2007), and assumes a crystalline homogeneity across cortical regions which are known to have substantial structural differences (e.g. V1 and IT cortices - Tsunoda et al. (2001)). Nonetheless, the level of detail is enough to generate testable predictions and guide future modelling and experimental work, while at the same time keeping the models open to future modifications and improvements through the incorporation of further biological details.

3.5 Original contributions in this chapter

- Review evidence suggesting the visual cortex can be understood in terms of a generative model, Bayesian inference and belief propagation.
- Provide a clear explanation of belief propagation in Bayesian networks, including an original illustrative toy network, with intuitive variables and causal relationships; and numerical step-by-step examples of the different types of evidence propagation.
- Compare analytically the two spiking neuron models of belief propagation in graphical models.
- Review and compare analytically the most relevant models of visual perception based on generative modelling approaches similar to belief propagation in Bayesian networks.
- Compare analytically several tentative cortical mappings of graphical models, and extract the main points of agreement and disagreement amongst them.

Chapter 4

Methods

This chapter describes in detail a theoretical and computational model which employs the mathematical tools described in Chapter 3, namely Bayesian networks and belief propagation, to simulate some of the anatomical and physiological properties of the ventral visual pathway, embodied in the HMAX model. Furthermore, the model tries to reproduce some of the observed phenomena described in Chapter 2, such as feedback modulation and illusory contour completion.

The chapter is organized as follows. Section 4.1 sums up the different layers and operations of the HMAX model and describes a how this model can be formulated as a probabilistic Bayesian Network implementing belief propagation. Section 4.2 specifies the exact network parameters of three different HMAX architectures and describes the corresponding Bayesian network that captures each set of parameters. Section 4.3 examines the learning methods used to generate the conditional probability tables of the Bayesian network, and how these weights approximately capture the original prototypes and operations of the HMAX model. Section 4.4 details how the selectivity and invariance operation of the HMAX model are approximated using the Bayesian belief propagation algorithm. Section 4.5 describes how feedback is implemented inherently in the proposed Bayesian network through the belief propagation algorithm. Additionally, it discusses the solutions implemented to deal with the problem of having multiple parents and loops in the network. Finally, Section 4.6 recapitulates and justifies the different approximations used by the model.

4.1 HMAX as a Bayesian network

4.1.1 HMAX model summary

The HMAX model (Riesenhuber and Poggio 1999, Serre et al. 2007b), which captures the basic principles of feedforward hierarchical object recognition in the visual system, has already been described in some detail in Section 2.1.2. This model was chosen as a starting point, firstly because it reproduces many anatomical, physiological and psychophysical data from regions V1, V4 and IT. The second reason is that it has been repeatedly argued that the main limitation of the HMAX model is that it does not account for the extensive feedback projections found in the visual cortex (Serre 2006, Walther and Koch 2007). Our proposed methodology, namely Bayesian networks and belief propagation, is ideal to tackle this problem and provide such an extension. Below is a brief technical outline of the different layers and operations in the original HMAX model, which will facilitate understanding of the proposed model. Figure 2.4 provides a graphical representation of the different layers in HMAX.

S1 layer - Units in this layer implement Gabor filters, which have been extensively used to model simple cell receptive fields (RF), and have been shown to fit well the physiological data from striate cortex (Jones and Palmer 1987). There are 64 types of units or filters, one for each of the $K_{S1}(=4)$ orientations $(0°, 45°, 90°, 135°)$ x $\Delta N_{S1}(=16)$ sizes or peak spatial frequencies (ranging from 7x7 pixels to 37 x 37 pixels, in steps of 2 pixels). The four different orientations and 16 different sizes, although an oversimplification, have been shown to be sufficient to provide rotation and size invariance at the higher levels. Phases are approximated by centring the Gabor filters at all locations. The RF size range is consistent with primate visual cortex ($0.2°$ to $1°$). The input image, a gray-valued image (160 x 160 pixels $\approx 5°x5°$ of visual angle) is filtered at every location by each of the 64 Gabor filters described by the following equation:

$$G_{x,y} = \exp\left(-\frac{(x\cos\theta + y\sin\theta)^2 + \gamma^2(-x\sin\theta + y\cos\theta)^2}{2\sigma^2}\right) \times \cos\left(2\pi\frac{1}{\lambda}(x\cos\theta + y\sin\theta) + \phi\right) \tag{4.1}$$

The parameters in the equation, that is, the orientation θ, the aspect ratio γ, the effective width

σ, the phase ϕ and the wavelength λ, determine the spatial receptive field of the S1 units. These parameters were adjusted so that the tuning profiles of S1 units match those of V1 parafoveal simple cells in monkeys (Serre and Riesenhuber 2004).

C1 layer - Units in this layer correspond to cortical complex cells showing a bigger RF size and a certain degree of position and size invariance. Each C1 unit receives input from a $\Delta N_{C1} \times \Delta N_{C1}$ square array of retinotopically organized S1 units with the same orientation, thus preserving feature specificity. C1 units are arranged in 8 scale bands, where units at each scale band pool from two S1 RF sizes, e.g. C1 scale band 1 pools from S1 units with RF sizes 7 and 9. The pooling grid size, ΔN_{C1}, ranges from 8 pixels to 22 pixels, in steps of 2 pixels, according to the C1 scale band. The pooling operation used is the *max* operation, i.e. the activity of each C1 unit is determined by the strongest of its ($\Delta N_{C1} \times \Delta N_{C1}$ positions $\times$ 2 RF sizes) afferent S1 units. This is shown in the following equation:

$$C1_{b_{C1},x_{C1},y_{C1},k} = \max_{b_i,x_i,y_i} \left(S1_{\{b_i,x_i,y_i\},k}\right) \tag{4.2}$$

where k represents the feature (in this case the filter orientation),

b_{C1}, x_{C1}, y_{C1} represents the band and location of the C1 unit,

$\{b_i, x_i, y_i\}$ represents the band and location of the afferent S1 units, and are given, as a function of the C1 unit's band and location and the network parameters, by the following expressions:

$$b_i \in \{2 \cdot b_{C1} - 1 \ , \ 2b_{C1}\} \tag{4.3}$$

$$x_i \in \{1 + (x_{C1} - 1) \cdot \varepsilon_{C1}(b_{C1}), \cdots, 1 + (x_{C1} - 1) \cdot \varepsilon_{C1}(b_{C1}) + \Delta N_{C1}(b_{C1})\} \tag{4.4}$$

$$y_i \in \{1 + (y_{C1} - 1) \cdot \varepsilon_{C1}(b_{C1}), \cdots, 1 + (y_{C1} - 1) \cdot \varepsilon_{C1}(b_{C1}) + \Delta N_{C1}(b_{C1})\} \tag{4.5}$$

This means each C1 unit represents a Gabor-like feature of the same orientation as the S1 units that feed into it, but with a certain position and size invariance. Additionally, C1 units

implement contrast invariance, mimicking complex cells in striate cortex, by taking the absolute value of their S1 inputs. Therefore, at each C1 location there are 32 C1 units, one for each of the $K_{C1}(=4)$ orientations $\times$ 8 scale bands. Note that, unlike S1 units, C1 units are not computed at every possible location but are sampled every ε_{C1} pixels or S1 units, where ε_{C1} ranges from 3 pixels to 15 pixels, in steps of 2 pixels, according to the C1 scale band.

Physiological data on simple and complex RF size, spatial frequency and orientation bandwidth are in good agreement with the model S1 and C1 tuning properties, as well as with the hypothesis of complex cells performing a *max* operation over simple cell afferents (Serre and Riesenhuber 2004).

S2 layer - The response of each S2 unit depends in a Gaussian-like way on the Euclidean distance between the input and previously learned prototypes. More specifically, it implements a Radial Basis Function (RBF) network, where the prototypes are the RBF centres. During the training phase, K_{S2} prototypes are learned from the C1 layer, each one composed of $\Delta N_{S2} \times \Delta N_{S2} \times K_{C1}(=4)$ elements. In some HMAX versions (Serre et al. 2007c) $K_{S2} = 2000$ and $\Delta N_{S2} = 3$, which yields 2000 prototypes with $3 \times 3 \times 4 = 36$ elements; while other implementations (Serre et al. 2007b) use values of $K_{S2} = 1000$ and ΔN_{S2} in the range $\{4, 8, 12, 16\}$. In summary, at each S2 location there are K_{S2} S2 units coding each of the learned prototypes.

During the recognition phase, the response of an S2 unit at a particular location and coding a specific learned prototype or RBF centre is calculated as the distance between the input patch of $\Delta N_{S2} \times \Delta N_{S2}$ C1 units, and the k^{th} stored prototype P_k, such that,

$$S2_{b_{S2},x_{S2},y_{S2},k} = \exp\left(-\beta \cdot \left|\left|C1_{\{b_i,x_i,y_i\}} - P_k\right|\right|^2\right) \tag{4.6}$$

where β is the square of the inverse width of the RBF and therefore defines the sharpness of the tuning curve,

b_{S2}, x_{S2}, y_{S2} represents the band and location of the S2 unit,

$\{b_i, x_i, y_i\}$ represents the band and location of the afferent C1 units, and is given, as a function of the S2 unit's band and location and the network parameters, by the following expressions:

$$b_i = B_{S2}$$

$$x_i \in \{x_{S2}, \cdots, x_{S2} + \Delta N_{S2}\} \tag{4.7}$$

$$x_i \in \{y_{S2}, \cdots, y_{S2} + \Delta N_{S2}\} \tag{4.8}$$

C2 layer - In the C2 layer, units perform the *max* operation pooling over a $\Delta N_{C2} \times \Delta N_{C2}$ square lattice of S2 units tuned to the same preferred stimulus, i.e. the same learned prototype. C2 units are therefore selective to the same stimulus as their S2 input units but present an increased position invariance. At each location, C2 units will code each of the $K_{C2} = K_{S2}$ learned prototypes, which can now be considered position invariant prototypes. In some HMAX versions (Serre et al. 2007c) ΔN_{C2} is set such that a single C2 unit for each prototypes receives input from S2 units at all locations and scale bands tuned to the same prototype. Other HMAX implementations (Serre et al. 2007b) employ values similar to those of the C1 layer, such that ΔN_{C2} takes the values $\{8, 12, 16, 20\}$, and the shift between S2 units, ε_{C2} takes the values $\{3, 7, 10, 13\}$, for each of the 4 C2 scale bands. Analogously to the C1 layer, each C2 scale band pools from two of the S2 scale bands, achieving size invariance in the C2 responses. It has been shown that the S2-C2 hierarchy produces both selectivity and invariance parameters that match observed responses in V4 (Cadieu et al. 2007).

S3 and C3 layers - For Serre et al. (2007c), S3 constitutes the top layer of the model. However, Serre et al. (2007b) implement two extra layers, C3 and S4. In this version of the model, the response of S3 units is based on a Radial Basis Function operation which computes the distance between patches of $\Delta N_{S3} \times \Delta N_{S3}$ C2 units and the K_{S3} previously stored prototypes of the same dimensions, analogous to the computation performed by S2 units (see Equation (4.6)). Finally, C3 units are obtained by performing the *max* operation over all S3 units tuned to the same prototype at all of the different spatial positions and scale bands. This leads to $K_{C3} = K_{S3}$ C3 units, coding each of the feature prototypes but with larger spatial invariance, so that if the

feature is present at any position in the input image, the corresponding C3 unit will elicit a response.

The top layer, S4 in Serre et al. (2007b) and S3 in Serre et al. (2007c), is implemented as a support vector machine (SVM) that uses the K_{C3} or K_{C2} output features in each case, to learn, in a supervised manner, the objects or input images. Using a training set of images, the weights of the support vector machine are adjusted in order to classify the output C2/C3 features generated by the input images into the different learned object categories.

Although the number of layers varies among previous versions of HMAX, in all cases the top layer tries to simulate the higher regions of the ventral path. Top level units present bigger RFs and are tuned to complex composite invariant features, which are consistent with the so-called view-tuned cells present in the higher levels of the ventral pathway, such as the infero-temporal cortex (Serre et al. 2007b, Quiroga et al. 2005, Hung et al. 2005).

Note that learning occurs in a developmental-like manner, meaning that weights are obtained from snapshots of activity patterns falling on the receptive field of units, which are then generalized across scales and positions (Serre 2006, Masquelier et al. 2007, Masquelier and Thorpe 2010). This is described in more detail in Section 2.1.2.

4.1.2 Probabilistic interpretation of HMAX: conversion to a Bayesian network

Chapter 3 has illustrated how Bayesian networks can model a wide variety different scenarios and facilitate probabilistic reasoning under conditions of uncertainty. The key to doing this is to correctly capture the structural and causal relationships between the factors involved in the target scenario to be modelled. The target scenario is object perception in the visual system, and for that reason I have developed a Bayesian network that captures the structural and causal relationships in the HMAX model. It is important to draw a line between the previous subsection, which describes the properties of an existing object recognition model (HMAX), and the rest of this chapter, which describes the methodology employed to develop a Bayesian network (with belief propagation) that reproduces the structure and functionality of the HMAX model, and extends it to include feedback processing.

The first step in this process is to define the equivalences between the HMAX model and the proposed Bayesian network. These are summed up in Figure 4.1 and are as follows:

1. Each node of the Bayesian network represents a specific location, band and layer of the HMAX model.

2. The discrete states of each node of the Bayesian network represent the different features coded at that location, band and layer of the HMAX model. For example each Bayesian node at layer S1 will have $K_{S1}(=4)$ features, representing the four different filter orientations of HMAX.

3. The discrete probability distribution over the states of each Bayesian node represents the sum-normalized responses of the HMAX units coding the different features at that location, band and layer. Therefore, the probability distribution of each node comprises the response of K HMAX units, where K is the number of different features at that layer.

4. The conditional probability tables (CPTs) that link each node in the Bayesian network with its parent nodes in the layer above represent the prototype weights used to implement selectivity in the HMAX model. Additionally, the CPTs are used to approximate the *max* (invariance) operation between simple and complex layers of the HMAX model. Learning the appropriate CPT parameters allows the model to approximate the HMAX functionality during the inference stage (using belief propagation) of the Bayesian network. This is described in further detail in Section 4.3.

Each node in the network implements the belief propagation algorithm, which has been described in detail in Section 3.3.3. Figures 4.2 and 4.3 show the specific operations implemented by each node, in the case of a single parent structure and a multiple parent structure, respectively. The former corresponds to a particularization of the latter. The operations performed correspond to Equations (3.27) to (3.31). The diagrams illustrate how to effectively implement belief propagation in a local and distributed manner. Note that to do this the top-down output messages of the node are made equivalent to the belief of the node. Therefore, the incoming

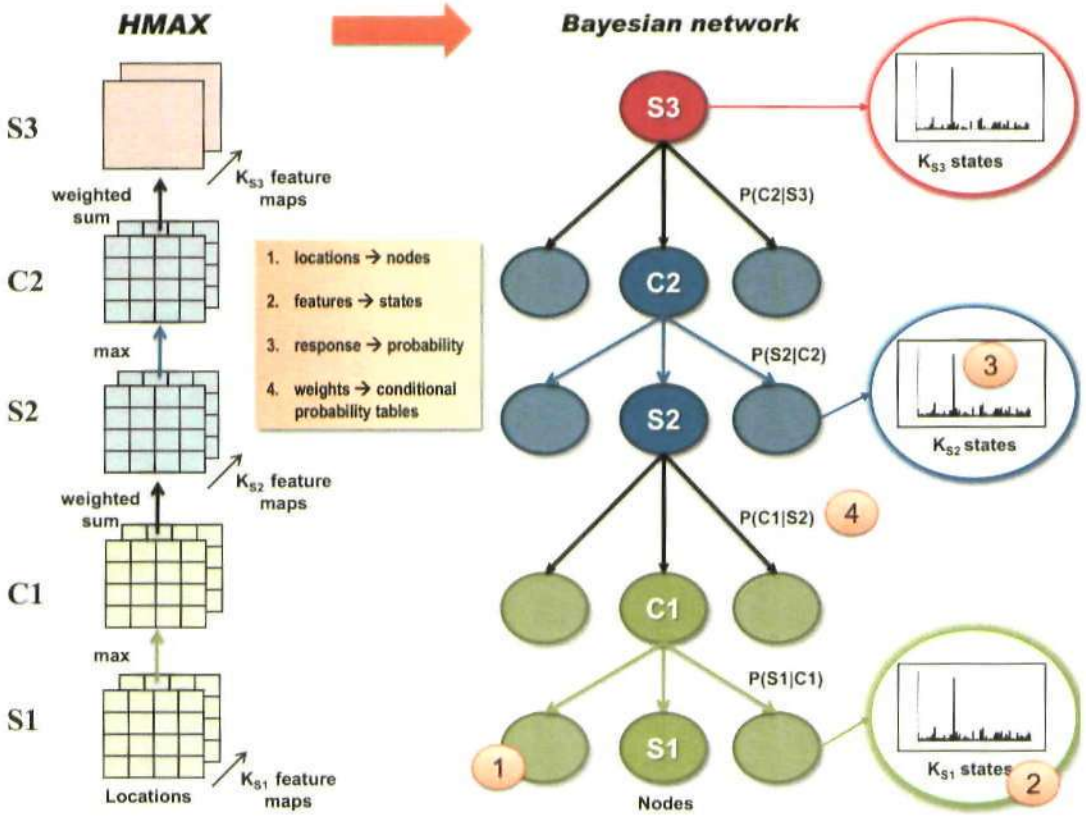


Figure 4.1: Probabilistic interpretation of HMAX as a Bayesian network. *Left*) Schematic representation of the HMAX model. At each layer, the response of each unit codes the presence of a specific feature at a given location. The invariance (*max*) and selectivity (weighted sum) operations are implemented in alternating layers. Right) Bayesian network representing the HMAX model network on the left: 1) each node represents a specific location, band and layer of the HMAX model; 2) the states of each node represent the different features; 3) the probability distribution of each node represents the sum-normalized response of the K HMAX units at that location; and 4) the conditional probability tables linking nodes of different layers represent the weights of the HMAX selectivity operation, as well as serving to approximate the HMAX invariance operation. The equivalences are summarized in the central orange box and are labelled with orange numbered circles over the resulting Bayesian network.

top-down messages to a node need to be divided by the output λ message to obtain the corresponding π message from the node above. This solution was proposed by Pearl (1988) and it avoids calculating the specific π messages for each child node. Instead it is more effective to simply feed back the belief and let each child node calculate its own input π message. As will be described later on, in cases where the total number of incoming messages is relatively high, the π message can simply be approximated by the belief.

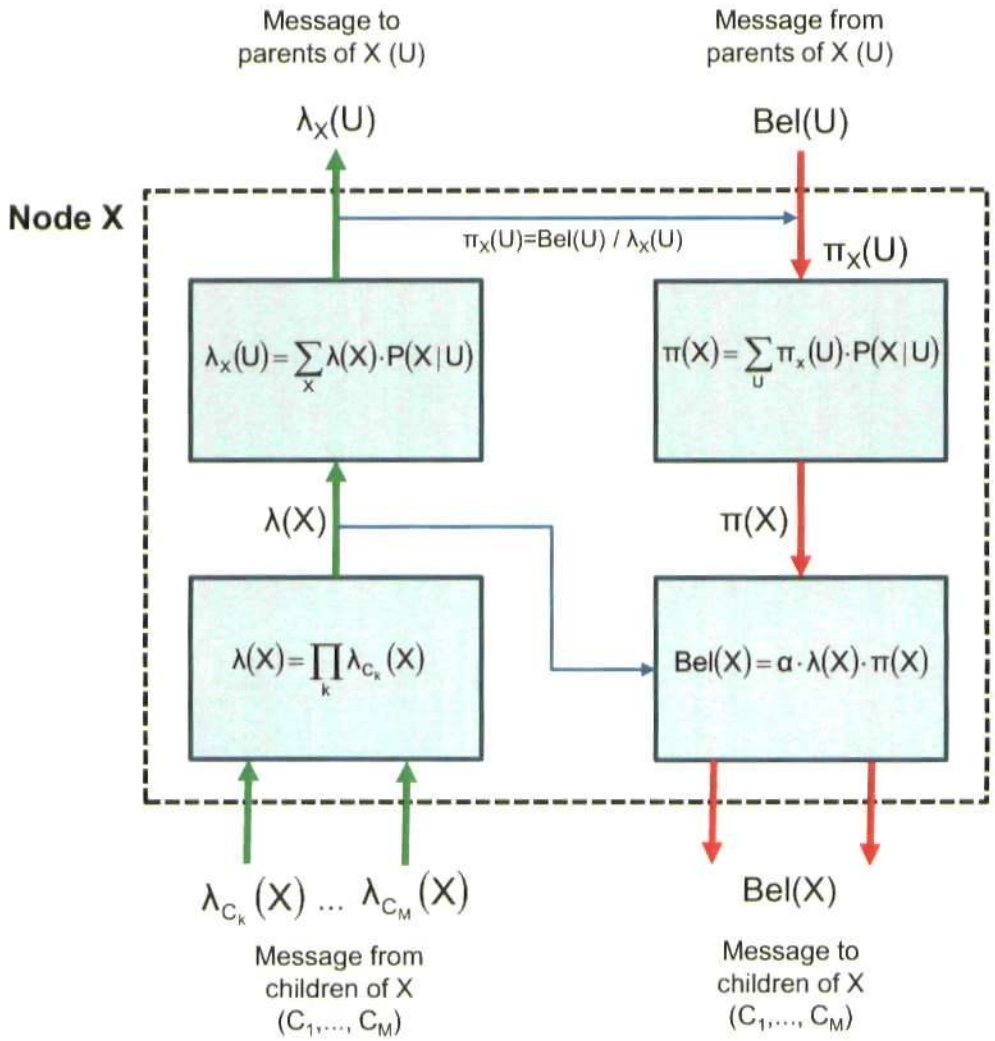


Figure 4.2: Internal structure of a node implementing belief propagation in a Bayesian network with a tree structure (one parent per node). Each node combines the bottom-up λ messages from child nodes with the top-down π message from the parent node to calculate the belief and the output λ message to the parent node. Note that the top-down output messages of the node are made equivalent to the belief of the node. Therefore, input top-down messages to a node need to be divided by the output λ message to obtain the corresponding π message from the node above (Pearl 1988). The operations described correspond to a particularization of Equations (3.27) to (3.31) for the single parent case. For more details see Section 3.3.3.

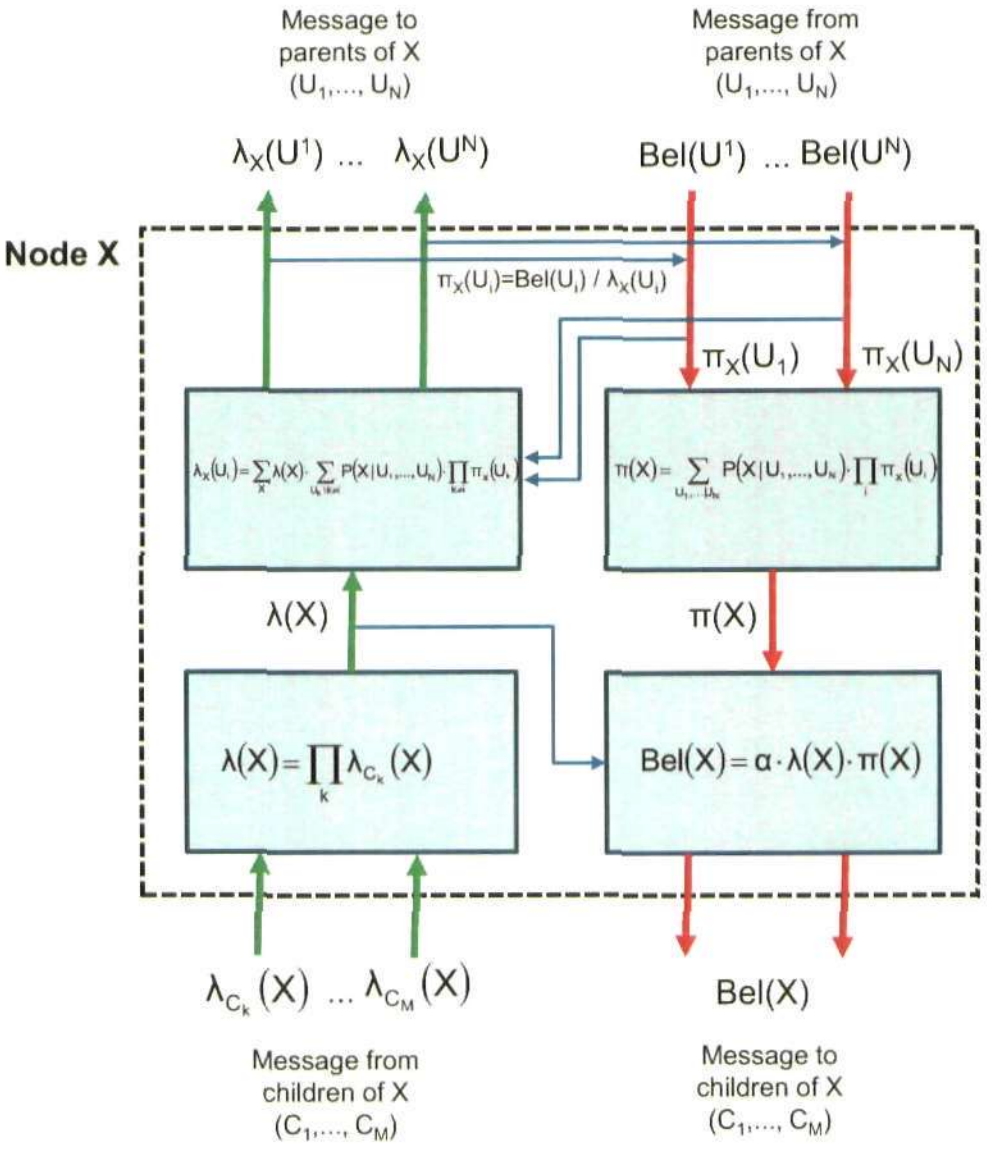


Figure 4.3: Internal structure of a node implementing belief propagation in a Bayesian network with a polytree structure (more than one parent per node). Each node combines the bottom-up λ messages from child nodes with the top-down π messages from the parent nodes to calculate the belief and the output λ messages to the parent nodes. Note that the top-down output messages of the node are made equivalent to the belief of the node. Therefore, input top-down messages to a node need to be divided by the output λ message to obtain the corresponding π message from the node above (Pearl 1988). The operations performed correspond to Equations (3.27) to (3.31), and were described in detail in Section 3.3.3.

4.2 Architectures

This section describes the specific structure parameters of the three different Bayesian networks employed. They are based on the parameters of two published versions of the HMAX mode (Serre et al. 2007c,b). For simplicity I have used the same parameter notation employed in these papers. Note that these parameters can be used to build the HMAX network as well as the functionally equivalent Bayesian network, as they define the topology of the network, i.e. the structure and the interconnectivity between the different elements of the network.

The first two layers of the network are equivalent in all three architectures and their parameters are summed up in Table 4.1.

4.2.1 Three-level architecture

The parameters for layers S2, C2 and S3 (Serre et al. 2007c) are shown in Table 4.2 and the resulting Bayesian network is illustrated in Figure 4.4. The number of nodes at each layer and band depend on the size of the input image and the pooling (ΔN) and sampling (ε) parameters. The figure shows the total number of nodes at each layer, assuming an input image of 160x160 pixels and the parameters defined by Table 4.2.

Due to inherent properties of Bayesian networks, each node in the graph can only have a fixed number of afferent nodes. For this reason, in order to obtain S2 nodes with features of different RF sizes, $\Delta N_{S2} = 4, 8, 12, 16$, these are implemented using separate nodes. Therefore, all S2, C2 and S3 nodes are repeated four times, one for each of the RF sizes. This is not illustrated in Figure 4.4 because the structure of each of the four sets of nodes is equivalent (as a function of

S1 parameters								
RF size, ΔN_{S1}	7, 9	11, 13	15, 17	19, 21	23, 25	27, 29	31, 33	35, 37
S1 types, K_{S1}	4 (0°;45°;90°;135°)							
C1 parameters								
Scale band	1	2	3	4	5	6	7	8
Grid size, ΔN_{C1}	8	10	12	14	16	18	20	22
Sampling, ε_{C1}	3	5	7	8	10	12	13	15
C2 types, K_{C1}	4 (0°;45°;90°;135°)							

Table 4.1: Comparison between two implementations using spiking neurons of graphical models and belief propagation

S2 parameters				
Scale band	1	2	3	4
RF size[a], ΔN_{S2}	4	8	12	16
S2 types, K_{S2}	1000			
C2 parameters				
Band pooling, ΔS_{C2}	All bands: $1 \cdots 8$			
Grid size, ΔN_{C2}	All S2 units			
C2 types, K_{C2}	1000			
S3 parameters				
RF size[b], ΔN_{S3}	1			
S3 types, K_{S3}	60			

[a] S2 prototype elements= $\Delta N_{S2} \times \Delta N_{S2} \times K_{C1}$ (4 orientations). Same for all scale bands.
[b] S3 prototype elements= $\Delta N_{S3} \times \Delta N_{S3} \times K_{C2}$ (1000 features).

Table 4.2: Parameters of the 3-layer architecture. Based on Serre et al. (2007c).

the corresponding parameters), and the nodes of different S2 RF sizes do not interact with each other.

The number of nodes in each S2 set is 2253 for $\Delta N_{S2} = 4$, 1572 for $\Delta N_{S2} = 8$, 1098 for $\Delta N_{S2} = 12$ and 758 for $\Delta N_{S2} = 16$. Bigger ΔN_{S2} imply less resulting S2 units as the number of S2 units is equal to the number of C1 units divided by ΔN_{S2}. The number of features of each RF size, K'_{S2} is set to the total number of features in layer S2 divided by four i.e. $K'_{S2} = K_{S2}/4 = 1000/4 = 250$.

Each node in the network has an associated CPT which links it with its parent nodes. Similarly, each node performs the same internal operations, shown in Figure 4.3, which correspond to the distributed implementation of belief propagation.

4.2.2 Alternative three-level architecture based on Yamane et al. (2006)

The parameters of this architecture are shown in Table 4.3 and the resulting Bayesian network is illustrated in Figure 4.5 (only from layer S2 above, as layers S1 and C1 are equivalent to the previous version). This architecture was introduced to try to improve the recognition of the *translated* dataset of input objects. It is a variation of the 3-level HMAX model (Serre et al. 2007c), with a reduced pooling range (RF size) at the top layers C2 and S3. More specifically, C2 prototypes do not pool over the whole set of S2 units, but over a smaller range (50% of the S2 map length), which leads to a 3-by-3 C2 grid of units. This then allows the S3 RF size to be set

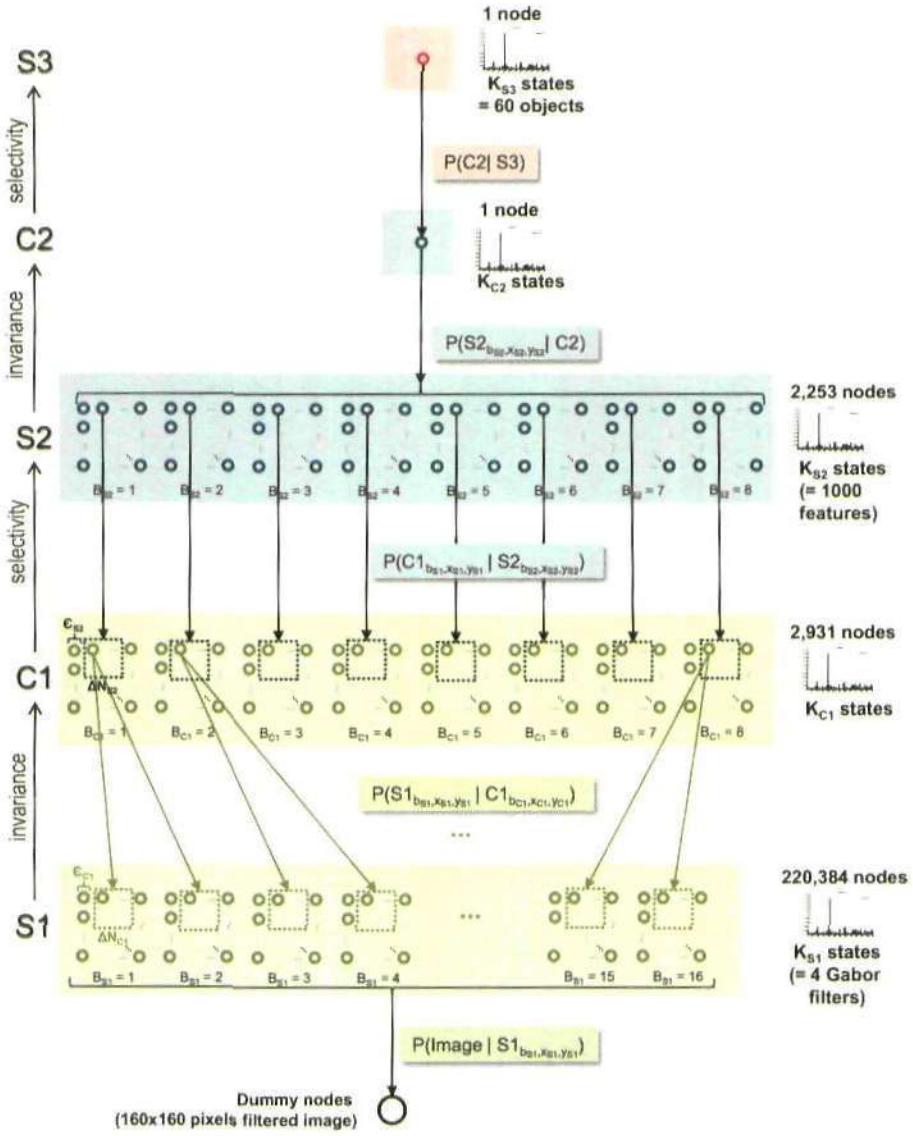


Figure 4.4: Bayesian network reproducing the structure and functionality of the 3-level HMAX model (Serre et al. 2007c). The number of nodes at each layer and band depend on the size of the input image and the pooling (ΔN) and sampling (ϵ) parameters. The figure shows the total number of nodes at each layer, assuming an input image of 160x160 pixels and the parameters defined by Table 4.2. Each node in the network has an associated CPT which links it with its parent nodes. Similarly, each node performs the same internal operations, shown in Figure 4.3, which correspond to the distributed implementation of belief propagation.

S2 parameters				
Scale band	1	2	3	4
RF size[a], ΔN_{S2}	4	8	12	16
S2 types, K_{S2}	1000			
C2 parameters				
Band pooling, ΔS_{C2}	All bands: $1 \cdots 8$			
Grid size, ΔN_{C2}	$0.5\times$ total S2 units			
Sampling, ε_{C2}	$0.25\times$ total S2 units			
C2 types, K_{C2}	1000			
S3 parameters				
RF size[b], ΔN_{S3}	2			
S3 types, K_{S3}	$2 \times 2 \times 60 = 240$			

[a] S2 prototype elements= $\Delta N_{S2} \times \Delta N_{S2} \times K_{C1}$ (4 orientations). Same for all scale bands.
[b] S3 prototype elements= $\Delta N_{S3} \times \Delta N_{S3} \times K_{C2}$ (1000 features).

Table 4.3: Parameters of the alternative 3-layer architecture based on Yamane et al. (2006).

to just 2×2 C2 units, and to learn 2×2 prototypes (one for each location) per object category, which leads to greater position invariance at the top layer. As a consequence, the learned high-level prototypes of objects contain some information about the spatial arrangement of their constituent parts, in agreement with the results shown by Yamane et al. (2006). This was also discussed in Section 2.1.1 and illustrated in Figure 2.3b.

Additionally, the smaller pooling ranges of each unit help to reduce the large fan-in of λ messages and to increase the specificity of feedback. All feedback results presented in the thesis are based on this architecture.

4.2.3 Four-level architecture

This architecture is based on the version of HMAX described in Serre et al. (2007b), and includes two extra layers. The parameters of this architecture are shown in Table 4.4 and the resulting Bayesian network is illustrated in Figure 4.6 (only layers above S2, as layers S1 and C1 are equivalent those shown in Figure 4.4). The main advantage of this architecture is the further processing by the two extra layers with smaller pooling ranges which leads to greater position and scale invariance and can increase selectivity in highly detailed images. However, these two extra layers also lead to greater complexity and higher approximation errors when implementing the model as a Bayesian network. For this reason, the four-level architecture was

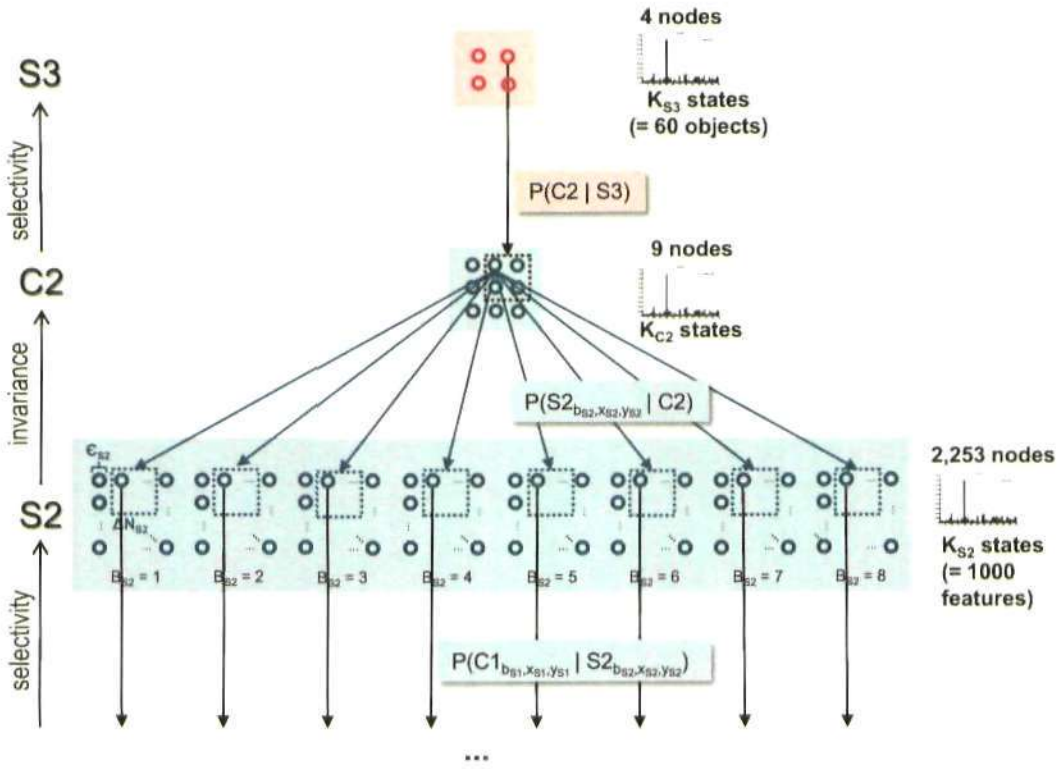


Figure 4.5: Bayesian network reproducing the structure and functionality of a modified version of the 3-level HMAX model (Serre et al. 2007c). The variation consists of a reduced pooling range (RF size) at the top layers C2 and S3 and was introduced to try to improve the recognition of the *translated* set of input objects. Only those layers above S2 are shown, as layers S1 and C1 are equivalent to the previous version. The number of nodes at each layer and band depends on the size of the input image and the pooling (ΔN) and sampling (ε) parameters. The figure shows the total number of nodes at each layer, assuming an input image of 160x160 pixels and the parameters defined by Table 4.3. Each node in the network has an associated CPT which links it with its parent nodes. Similarly, each node performs the same internal operations, shown in Figure 4.3, which correspond to the distributed implementation of belief propagation.

S2 parameters				
RF size[a], ΔN_{S2}	3			
S2 types, K_{S2}	2000			
C2 parameters				
Scale band	1	2	3	4
Band pooling, ΔS_{C2}	1,2	3,4	5,6	7,8
Grid size, ΔN_{C2}	8	12	16	20
Sampling, ε_{C2}	3	7	10	13
C2 types, K_{C2}	1000			
S3 parameters				
RF size[b], ΔN_{S3}	3			
S3 types, K_{S3}	1000			
C3 parameters				
Band pooling, ΔS_{C3}	All bands: $1 \cdots 4$			
Grid size, ΔN_{C3}	All S3 units			
C3 types, K_{C3}	1000			
S4 parameters				
RF size[c], ΔN_{S3}	1			
S3 types, K_{S3}	60			

[a] S2 prototype elements= $\Delta N_{S2} \times \Delta N_{S2} \times K_{C1}$ (4 orientations). Same for all scale bands.
[b] S3 prototype elements= $\Delta N_{S3} \times \Delta N_{S3} \times K_{C2}$ (1000 features).
[c] S4 prototype elements= $\Delta N_{S4} \times \Delta N_{S4} \times K_{C3}$ (1000 features).

Table 4.4: Parameters of the 4-layer architecture. Based on Serre et al. (2007c) but with $K_{S2} = K_{S3} = 1000$ features, instead of 2000.

only used for comparison with the other architectures during the the feedforward recognition process.

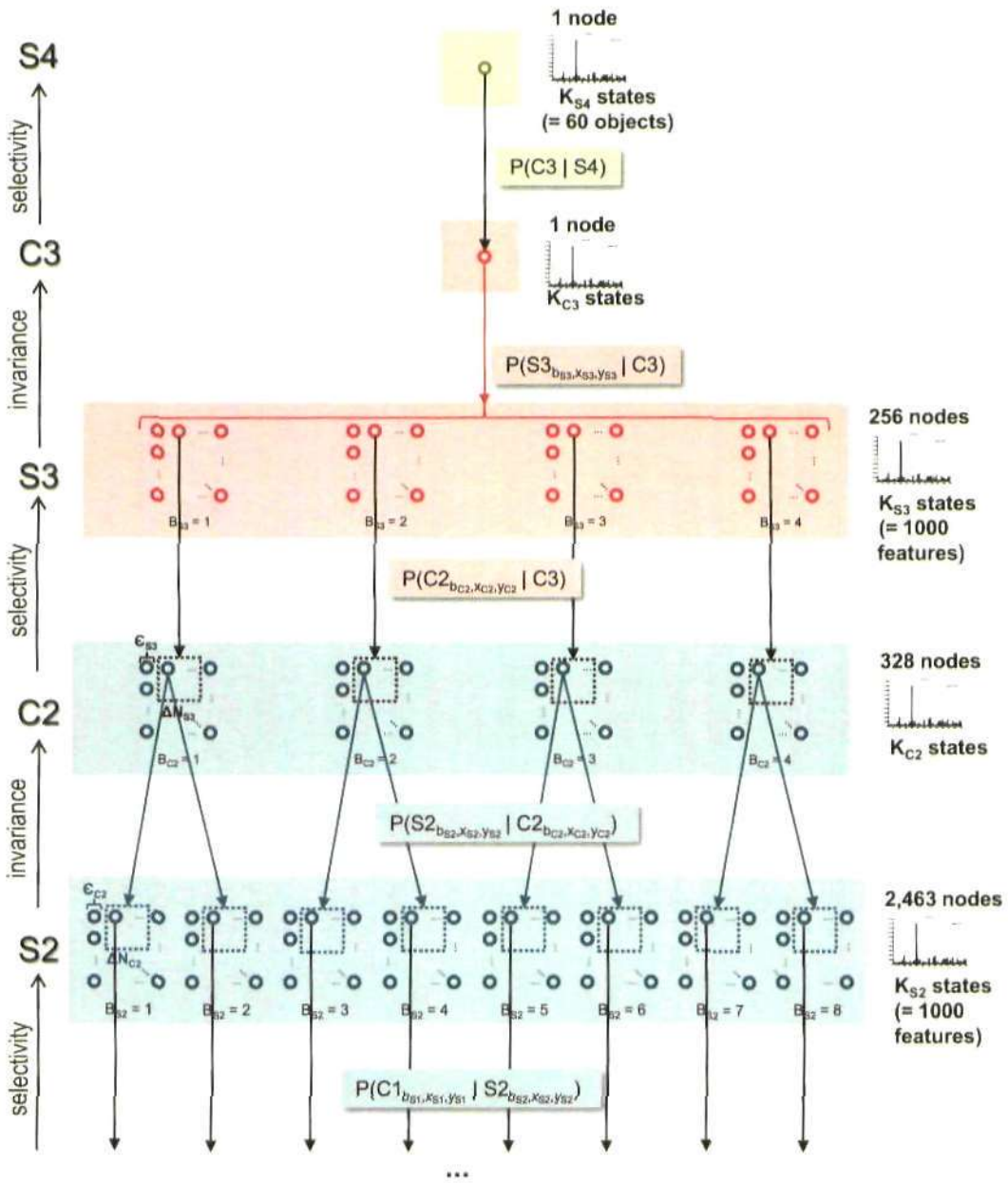


Figure 4.6: Bayesian network reproducing the structure and functionality of the 4-level HMAX model (Serre et al. 2007b). The number of nodes at each layer and band depend on the size of the input image and the pooling (ΔN) and sampling (ε) parameters. The figure shows the total number of nodes at each layer assuming an input image of 160x160 pixels, and the parameters defined by Table 4.4. Each node in the network has an associated CPT which links it with its parent nodes. Similarly, each node performs the same internal operations, shown in Figure 4.3, which correspond to the distributed implementation of belief propagation.

4.3 Learning

This section describes how to learn the conditional probability tables (CPTs) of each of the nodes in the Bayesian network in order to approximate the selectivity and invariance operations of the HMAX model. For this learning stage an important assumption is made in order to simplify the process. The network is assumed have a single parent per node (tree structure with no loops) so that the feedforward λ messages are not affected by the top-down feedback π messages. The bottom-up λ message from a node with a single parent does not include evidence from that parent (see Section 3.3.3 for details).

The reason for making this assumption is that the CPTs in the network are learned in an unsupervised manner, starting from the bottom layer (following HMAX learning methods), based on the response obtained at each layer. In order to calculate the response of nodes with multiple parents, the messages from all parents need to be combined using the CPTs that relate the node to its parents. However, these CPTs would still be unknown. This implies that, theoretically, in this type of network, all the CPTs would need to be learned at the same time. By assuming nodes with a single parent, the λ messages, based solely on bottom-up evidence, can be used as a reference to learn the appropriate weights layer by layer. Similar assumptions are made in other related models (Epshtein et al. 2008, George and Hawkins 2009, Hinton et al. 2006). The learning process is now described one layer at a time.

4.3.1 Image-S1 weights

The input image is pre-processed with a battery of Gabor filters described by Equation (4.1) with the parameter range described in Table 4.1, i.e. at 4 different orientations and 8 sizes. Each of the filters is applied at every location of the image. The filtered responses, normalized over the four orientations at each location and scale, are used as the output λ messages of a set of dummy nodes that feed onto the S1 nodes. As explained in Section 3.3.3, dummy nodes do not encode a variable or have a belief, but just generate λ messages for the parent nodes. For this reason there is no need to define the CPTs between the dummy nodes and the S1 nodes.

4.3.2 S1-C1 CPTs

The belief propagation operations between S1 and C1 need to approximate the *max* operation implemented in the original HMAX model. To do this we propose increasing the number of states of the C1 layer so that for each S1 orientation there are $K_{C1group}$ states coding different spatial arrangements of S1 units. In this way all the C1 states corresponding to the same S1 orientation can be grouped together and treated as a single state during the generation of the output λ message. The operation to compute the output λ message implements the sum over all the states of each C1 group. In other words, C1 nodes provide a distribution over S1 features and locations, which after marginalizing (summing) over the locations during the generation of the output λ message, provides an approximation to the *max* operation.

However, the number of different possible spatial arrangements of S1 units converging on a C1 unit is given by the number of k-combinations of the n-element set equal to the binomial coefficient, $\binom{n}{k}$, where $n = \Delta N_{C1} \cdot \Delta N_{C1} \cdot 2$ bands and k is the number of active units (assuming binary values). For example, for $n = 8 \cdot 8 \cdot 2 = 128$ and $k = 32$, the number of possible spatial arrangements is $\binom{128}{32} \approx 10^{32}$. This is just a lower bound on the real number of combinations, as we would need to sum over all the different values of k, and the weight values at each location are not necessarily binary, but range from 0 to 1. Creating a distribution for each C1 node containing $K_{C1} = K_{S1} \cdot K_{C1group} = 4 \cdot 10^{32}$ states, is obviously intractable.

For this reason, the value $K_{C1group}$ is limited to include only the most common arrangements of S1 units for each orientation. Figure 4.7 portrays a toy example of this method where the number of S1 units is $n = 3 \cdot 3 = 9$, the number of S1 states is $K_{S1} = 4$ (orientations), the number of features per group is $K_{C1group} = 3$ and the resulting number of C1 states is $K_{C1} = 4 \cdot 3 = 12$. The equation to calculate $\lambda(C1)$, which combines the bottom-up evidence, and $\lambda_{C1}(S2)$, which sends the bottom-up evidence to S2, are shown in the diagram.

Note that, as illustrated in the toy example, the weights are learned for each fixed C1 state=i as a function of the n afferent S1 units and the j S1 states per node. This yields a weight matrix (shown on the bottom left) for each of the C1 i states. However, the CPTs of a Bayesian network are defined as a function of the child and the parent states, j and i respectively, for each fixed

$S1_n$ node. Therefore, the original weight matrices are converted into one CPT per S1 node as shown at the bottom left of Figure 4.7.

The implementation in the real model follows that described in the toy example. The method employed to learn the $K_{C1group}$ most common arrangements of S1 units at each orientation feeding to a C1 unit is a clustering technique, namely the *k-means* algorithm (Bishop 1995). This method can be understood as a type of Expectation-Maximization algorithm. This clustering method is applied over a set of S1 patches obtained from the training input images. To select and pre-process the set of patches to be clustered for the i S1 state (orientation) or C1 group, the following steps are performed:

1. Select the next patch, $P_{potential}$ of size $n = \Delta N_{C1} \cdot \Delta N_{C1} \cdot 2$ (bands) from each state of the S1 nodes' λ response, i.e. $\lambda(S1_{\{b,x,y\}} = i)$, where $\{b,x,y\}$ represent the band and location of the S1 node and match the patch size, n; and i is a specific S1 orientation or C1 group.

2. Keep patch $P_{potential}$ only if its maximum value is above a threshold T_{min}, where T_{min} is given as a function of the maximum overall value of $\lambda(S1_{b,x,y} = i) \; \forall \; b,x,y$. Formally, if $max(P_{potential}) \geq T_{min}$ then $P_{selected} = P_{potential}$, where $T_{min} = \alpha \cdot max(\lambda(S1_{b,x,y} = i))$, and a typical value for α would be 0.9. This ensures that the weight matrix is calculated based on significant $\lambda(S1)$ responses which are close to the overall maximum response elicited by S1 nodes, and not based just on the local maximum of each patch.

3. For each selected patch, $P_{selected}$, keep only the values above T_{min} and set all other values to 0, i.e. $P_{selected} = floor(P_{selected}/T_{min})$. This ensures the weaker responses do not affect the calculation of the weight matrix.

[1]Caption for Figure 4.7. Toy example illustrating how to approximate the *max* operation using the CPTs between S1 and C1 nodes. The 12 states of the C1 node are organized in four groups each corresponding to one of the S1 states or orientations. Each of the $K_{C1group} = 3$ states within a C1 group codes a different spatial arrangement of the input S1 nodes. The weights between each of the C1 states and the S1 nodes are shown in the bottom-left tables. These are then converted to the corresponding CPTs between each S1 node and its parent C1 node as shown in the tables of the bottom-right. The belief propagation equations in the top-right square show how the CPTs are used to generate the output λ messages from each S1 node to its C1 parent node. These messages are then combined multiplicatively by the C1 likelihood function $\lambda(C1)$ and used to generate the output λ messages from each C1 node to its S2 parent node. In summary, C1 nodes provide a distribution over S1 features and locations, such that marginalizing (summing) over the locations during the generation of the output λ message to the parent S2 node, provides an approximation to the *max* of each S1 feature over the pooling region.

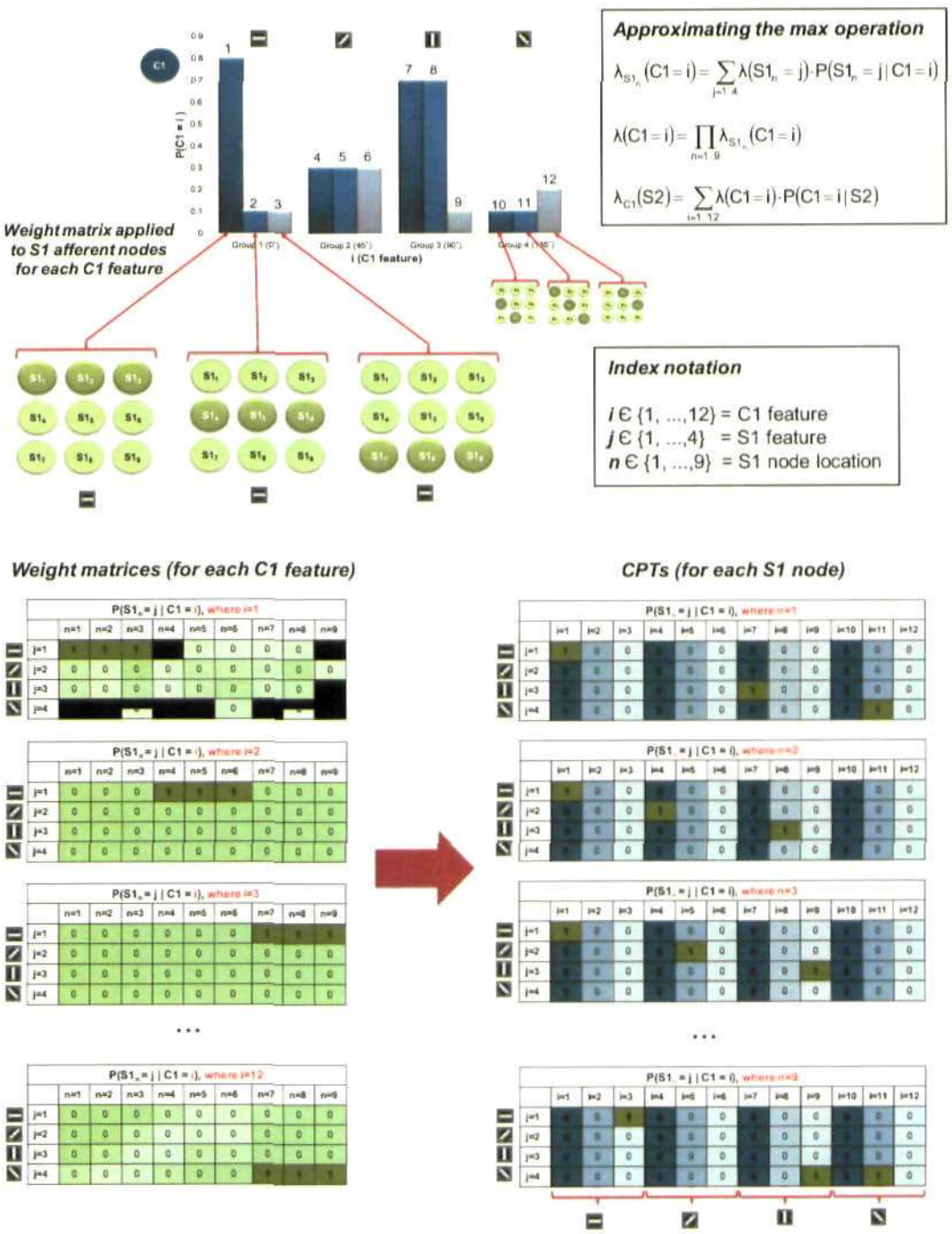


Figure 4.7: For caption see footnote[1].

The k-means clustering algorithm is then applied over the resulting selected patches. It works by generating initially $K_{C1group}$ random clusters. Then each of the $P_{selected,i}$ patches is assigned to the nearest cluster, and the value of the cluster is modified to reflect the mean value of all assigned patches. The resulting $K_{C1group}$ clusters for each orientation and C1 band represent the weight matrices between a C1 node and its $n = \Delta N_{C1} \cdot \Delta N_{C1} \cdot 2$ S1 afferent nodes.

After all the clusters or weight matrices have been calculated, the minimum number of non-zero elements in the weight matrices across all C1 groups is obtained. The number of non-zero elements of all weight matrices is set to this minimum, and the matrix elements are sum-normalized to one. This ensures that during the inference process the weights are balanced across the different C1 groups, and the result of the computation depends on the λ distribution. Otherwise if, for example, the C1 states corresponding to the horizontal orientation group had more non-zero elements, the $\lambda(C1)$ responses would be biased towards the horizontal orientation states. Note that an independent minimum value of non-zero elements is calculated for each scale band.

For some of the feedback results the number of non-zero elements of the S2-C2 weight matrix was increased to improve the S2 response reconstruction from the C2 feedback.

The resulting weight matrices, learned from the training dataset of 60 object silhouettes following the clustering procedure described, represent the $K_{C1group}$ most common arrangements of S1 units for each C1 group and scale band. These are shown in Figure 4.8 for a value of $K_{C1group} = 10$. The weights obtained show very clear and selective patterns which match what would be expected statistically from natural images, i.e. the arrangement of the S1 nodes tends to match the S1 orientation of the unit, which speaks for a coherence between the local and more global patterns. Note that each C1 node receives input from the S1 nodes at 2 scale bands, and these weights are represented adjacent to each other, e.g. weights for a C1 node in scale band 1 receive input from S1 nodes in scale bands 1 and 2, and are therefore shown as two 8×8 adjacent matrices = 16×8 elements.

The final step is to convert the weight matrices shown in Figure 4.8 to the corresponding CPTs of each S1 node $P(S1|C1)$, as was illustrated in the toy example in Figure 4.7. To conform to probability rules, each column of this table must add up to 1. This ensures that, for example,

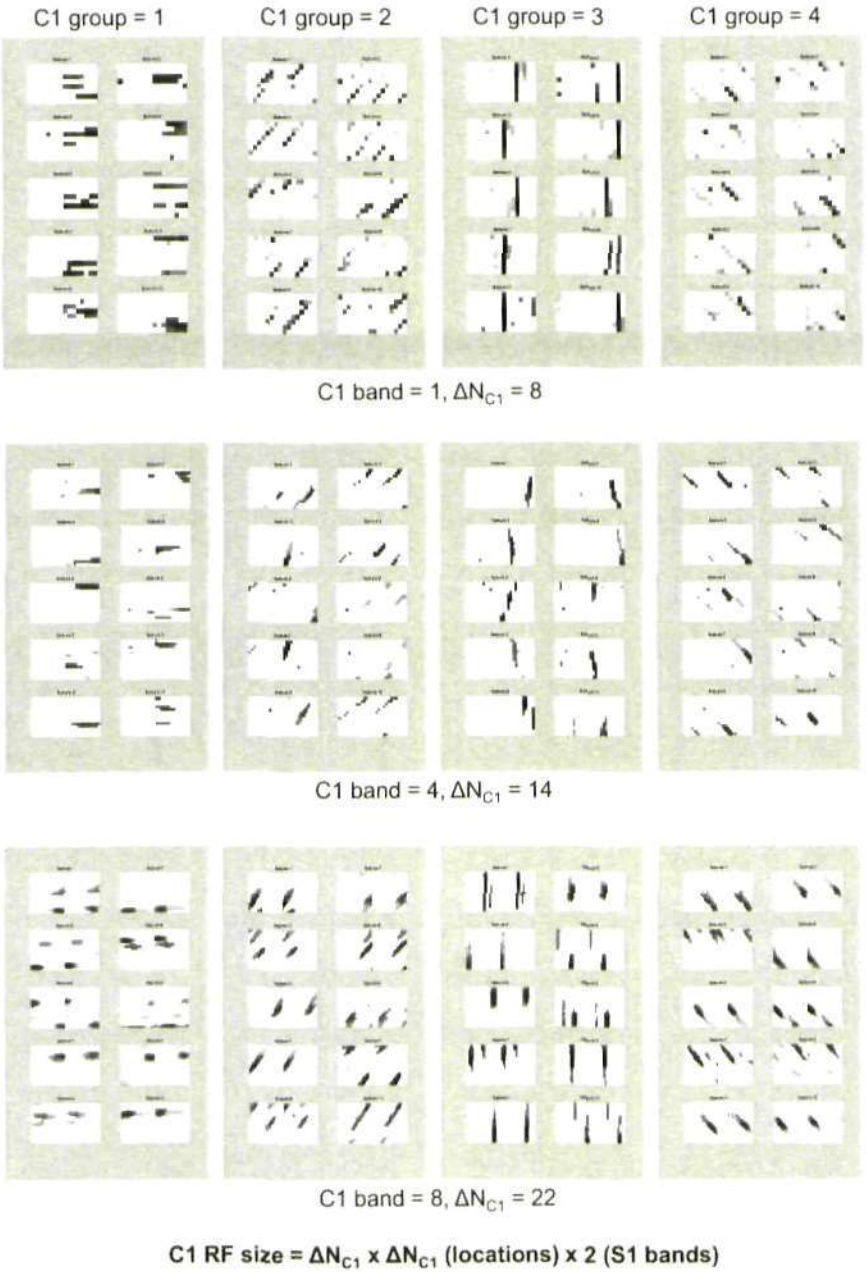


Figure 4.8: Weight matrices between a C1 node and its afferent S1 nodes. These are learned from the training dataset of 60 object silhouettes following the clustering procedure described, and represent the $K_{C1group} = 10$ most common arrangement of S1 nodes for each C1 group and scale band. Note, for each scale band the pooling range, ΔN_{C1}, varies. The weights obtained show very clear and selective patterns which match what would be expected statistically from natural images, i.e. the arrangement of the S1 nodes tends to match the S1 orientation of the unit, which speaks for a coherence between the local and more global patterns. Note that each C1 node receives input from the S1 nodes at 2 scale bands, and these weights are represented adjacent to each other, e.g. weights for a C1 node in scale band 1 receive input from S1 nodes in scale bands 1 and 2, and are therefore shown as two 8×8 adjacent matrices = 16×8 elements.

when all afferent S1 nodes have a flat λ distribution, as in blank regions of the image, the parent C1 node will also show a flat distribution.

In summary, the C1 layer becomes an intermediate step that converts combinations of S1 features and spatial arrangements into the states of a single C1 node. The *max* operation only occurs during the generation of the output λ messages to the S2 layers, which groups these states via the learned weight matrices. This method also provides a way to feed back information from complex to simple layers, where each complex feature corresponds to a specific arrangement of simple features. The method is equivalent to that employed by Hierarchical Temporal Networks (George and Hawkins 2009), where features in each node are combined into temporal groups or Markov chains. The method used here, however, preserves the Bayesian network structure by implementing the grouping of features in the weights of the CPTs.

4.3.3 C1-S2 CPTs

To learn the selectivity weights between layers C1 and S2, the *minimum distance* algorithm is employed. This algorithm was also used to extract the most common spatial patterns (equivalent to selectivity weights) in the Hierarchical Temporal Memory model (George and Hawkins 2009). In the HMAX model, the selectivity weights, or the prototypes which serve as centres for the Radial Basis Functions, were extracted at random from the C1 maps generated by the training images. However, in our model, the *minimum distance* algorithm provides better results, as it ensures the extracted prototypes maximize the Euclidean distance between each other. The algorithm works as follows:

1. All features, potential S2 prototypes $P_{potential}$, are extracted by sampling from all the locations and bands of the $\lambda(C1)$ response generated for each of the training images, i.e. $\lambda(C1_{b,x,y} = i)\ \forall\ b,x,y$. The number of elements for each prototype is $\Delta N_{S2} \times \Delta N_{S2} \times (K_{C1}/K_{C1group})$, i.e. the S2 RF size times the number of C1 states divided by the states per group. To learn the S2 prototypes, it is more efficient to obtain a single value for each C1 group by summing over all the features belonging to that group. In other words, although each C1 node is composed of 40 states, only 4 values, corresponding to the sum of each group, are used to compute the S2 prototypes.

2. The list of selected prototypes, $P_{selected}$, will initially contain no prototypes. A parameter called the *minimum distance*, D_{min}, is initialized to a relatively high starting value.

3. The algorithm loops through all the potential prototypes $P_{potential}$. Prototypes are added to the selected prototype list, $P_{selected}$, if the Euclidean distance to all previously stored prototypes is above the *minimum distance*, i.e. if $d(P_{potential,i}, P_{selected,j}) \geq D_{min}$ $j \in \{1..N\}$ then $P_{selected,N+1} = P_{potential,i}$, where N is the number of selected prototypes.

4. Lower D_{min} and repeat step 3 until $N = K_{S2}$. The initial value of D_{min} and the decreasing step size in each iteration dictate the dissimilarity between the final selections of prototypes.

The S2 prototypes represent the weight matrix between a parent S2 node and all of its C1 afferent nodes. The corresponding CPT for each C1 node, i.e. the weight matrix between each C1 node and all of its S2 parents $P(C1|S2)$, is calculated in an analogous way to that described for the S1-C1 CPTs. Note that the CPT elements for the C1 features within the same group are set to the same value. This is because the CPTs are derived from the weight matrices that contain a single value for each C1 group.

Figure 4.9 follows from the previous toy example where now the number of C1 units is $n = 3 \cdot 3 = 9$, the number of C1 states is $K_{C1} = 12$ and the number of S2 states (prototypes) is $K_{S2} = 10$. The equation to calculate $\lambda_{C1}(S2)$, which sends the bottom-up evidence to S2, and $\lambda(S2)$, which combines the bottom-up evidence, are shown in the diagram.

The toy example illustrates how the weights are learned for each fixed S2 state=i as a function of the n afferent C1 nodes and the j C1 states per node. This yields a weight matrix (shown on the bottom left) for each of the S2 i states. However, the CPTs of a Bayesian network are defined as a function of the child and the parent states, j and i respectively, for each fixed C1

[2]Caption for Figure 4.9. Toy example illustrating how to approximate the *selectivity* operation using the CPTs between C1 and S2 nodes. The weights between each of the S2 states and the C1 nodes are shown in the bottom-left tables. These are then converted to the corresponding CPTs between each C1 node and its parent S2 node as shown in the tables of the bottom-right. The belief propagation equations at the top-left square show how the CPTs are used to generate the output λ messages from each C1 node to its S2 parent node. These messages are then combined multiplicatively by the S2 likelihood function $\lambda(S2)$ and used to generate the output λ messages from each C1 node to its S2 parent node. Note that the CPT elements for the C1 features within the same group are set to the same value, as only one value is learned per C1 group.

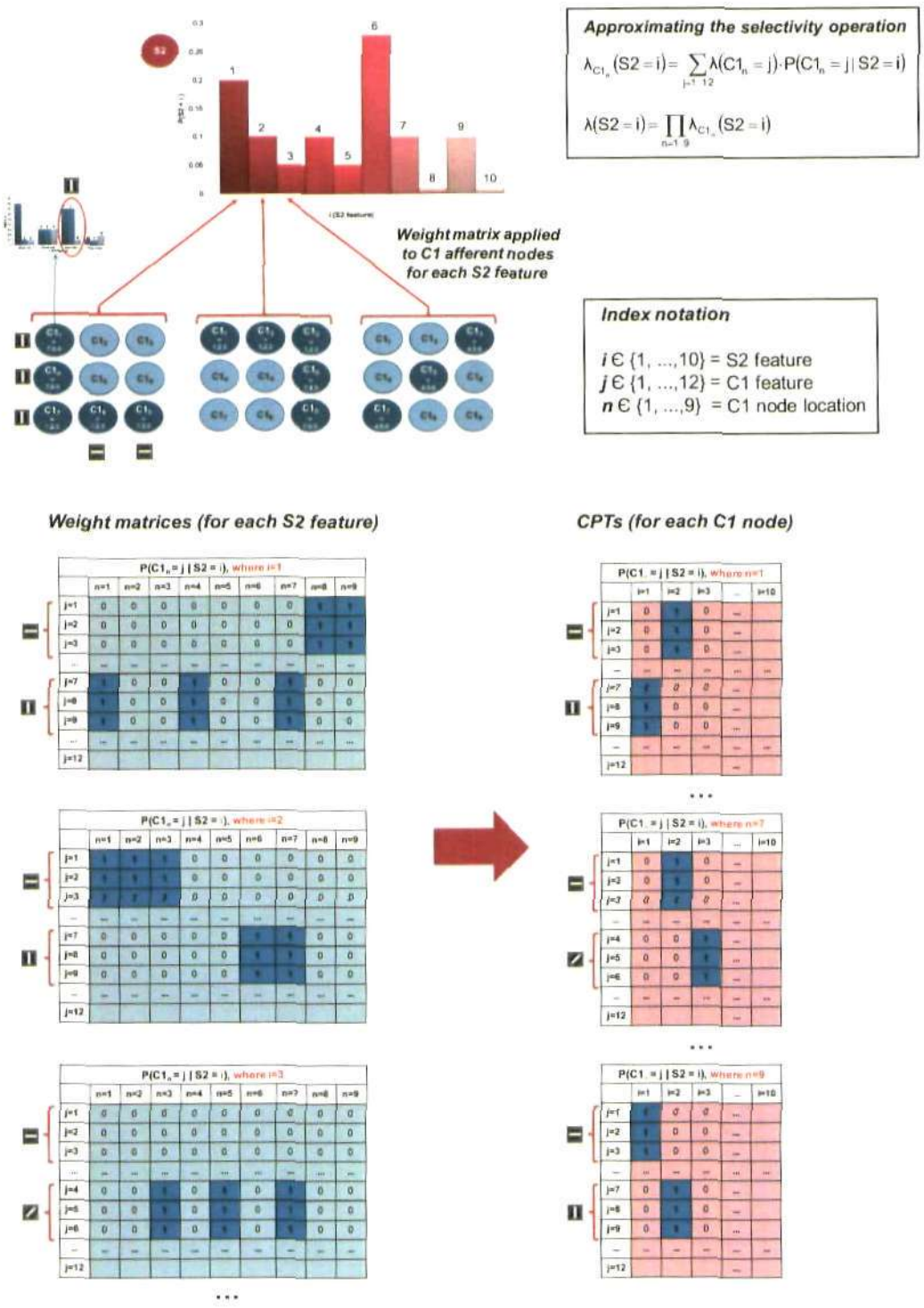


Figure 4.9: For caption see footnote[2].

node=n. Therefore, the original weight matrices are converted into one CPT per C1 node as shown at the bottom left of Figure 4.9.

Figure 4.10 shows a sample of 50 of the 250 S2 prototypes for each of the 4 RF sizes or ΔN_{S2} values. These have been extracted through using the *minimum-distance* algorithm described in this section and are shown before being converted to the CPTs. These prototypes are common for all scale bands. Each prototype represents the weights for each of the four orientations in a 2-by-2 grid of adjacent images, where top-left=0°,top-right=45°, bottom-left=90° and bottom-right=135°.

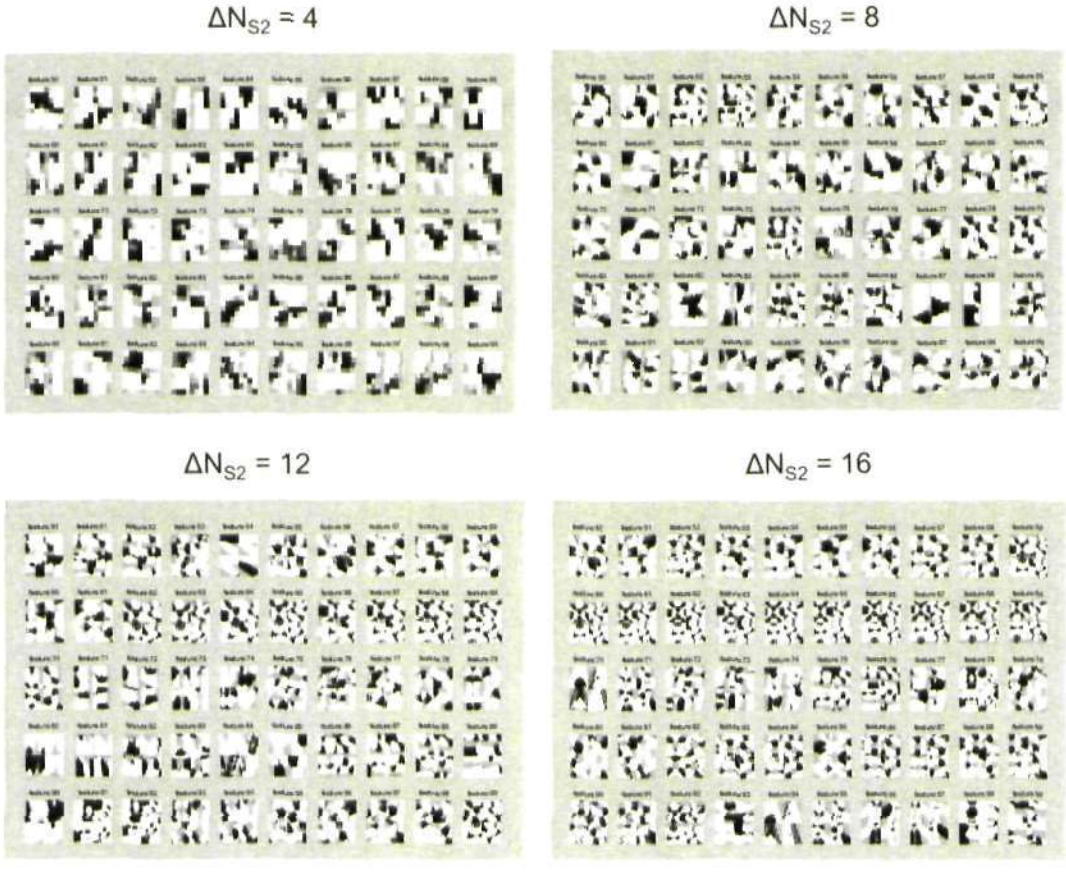


Figure 4.10: Weight matrices between an S2 node and its afferent C1 nodes. These are learned from the training dataset of 60 object silhouettes following the *minimum-distance* algorithm and are shown before being converted to the CPTs. The figure shows a sample of 50 of the 250 S2 prototypes for each of the 4 RF sizes or ΔN_{S2} values. These prototypes are common for all scale bands. Each prototype represents the weights for each of the four orientations in a 2-by-2 grid of adjacent images, where top-left=0°,top-right=45°, bottom-left=90° and bottom-right=135°.

4.3.4 S2-C2 CPTs

The weights between each C2 node and its afferent S2 nodes are learned using the same methodology described for the S1-C1 layers, i.e. k-means clustering to obtain the most common arrangements of S2 units. In this case the algorithm extracts $K_{C2group}$ clusters of size $N_{C2} \times N_{C2} \times N_{bands}$ for each C2 group or S2 state. The parameter N_{bands} represents the number of S2 bands being pooled from, and varies for the different architectures presented.

The resulting weight matrices, learned from the training dataset of 60 object silhouettes following the clustering procedure are shown in Figure 4.11 for a value of $K_{C2group} = 10$. Note that each C2 node receives input from the S2 nodes of up to 8 scale bands.

The weight matrices shown in Figure 4.11 are converted to one CPT per each S2 node using an equivalent procedure to that described for the S1-C1 CPTs.

4.3.5 C2-S3 CPTs

The weights between each S3 node and its afferent C2 nodes are learned in a supervised manner for each of the $K_{S3} = 60$ training images. For the 3-layer architecture, the $\lambda(C2)$ response for each training image becomes the prototype weight matrix. The CPT $P(C2|S3)$ containing $K_{C2}(= 10000) \times K_{S3}(= 60)$ elements can be easily obtained in this manner, by normalizing the weight matrices for each prototype. In other words, the prototype of each input image is learned as a function of the $\lambda(C2)$ response and converted to a CPT relating C2 and S3, as shown in Figure 4.11.

In the case of the alternative 3-layer architecture where there are 9 C2 nodes and 4 S3 nodes, the learning method is also supervised but the size and number of prototypes varies. The size of the S3 prototypes is now $\Delta N_{S3} \times \Delta N_{S3} = 2 \times 2$ C2 units; and these are learned from the 4 possible locations within the C2 units (see top of Figure 4.5), leading to $K_{S3} = 60$ objects $\cdot$ 4 locations =

[3]Caption for Figure 4.11. Weight matrices between a C2 node and its afferent S2 nodes. These are learned from the training dataset of 60 object silhouettes following the clustering procedure described, and represent the $K_{C2group} = 10$ most common arrangement of C1 nodes for each C2 group and scale band. Note that each C2 node receives input from the S2 nodes of up to 8 scale bands, where, for each scale band, the pooling range, ΔN_{C2}, is different. Similarly, the weight matrices are divided according to the S2 RF sizes, as the S2 response maps for S2 RF size, have different sizes and yield different S2-C2 weights. For purposes of clarity, a single C2 feature is highlighted using a red dotted ellipse.

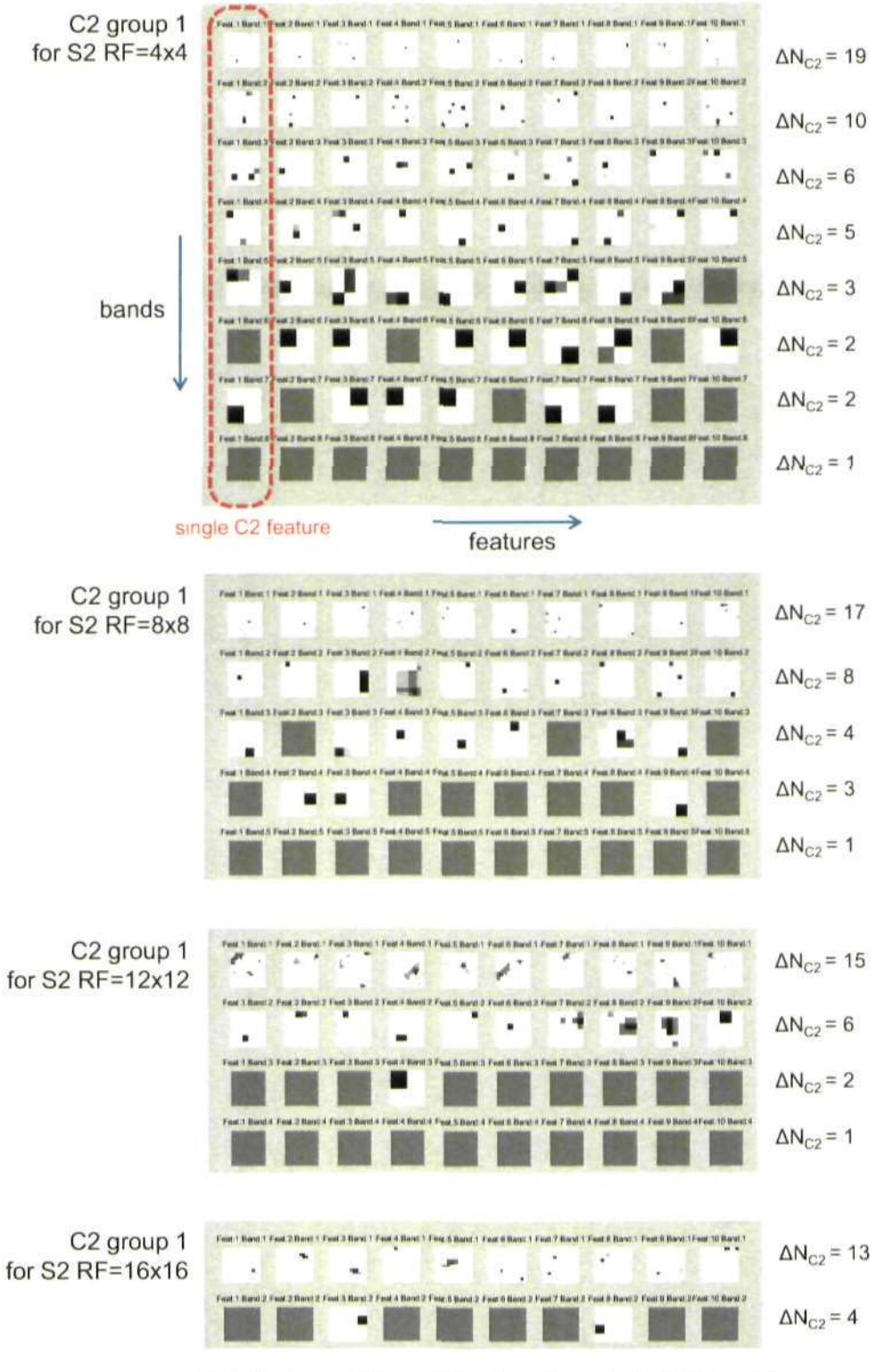


Figure 4.11: For caption see footnote[3].

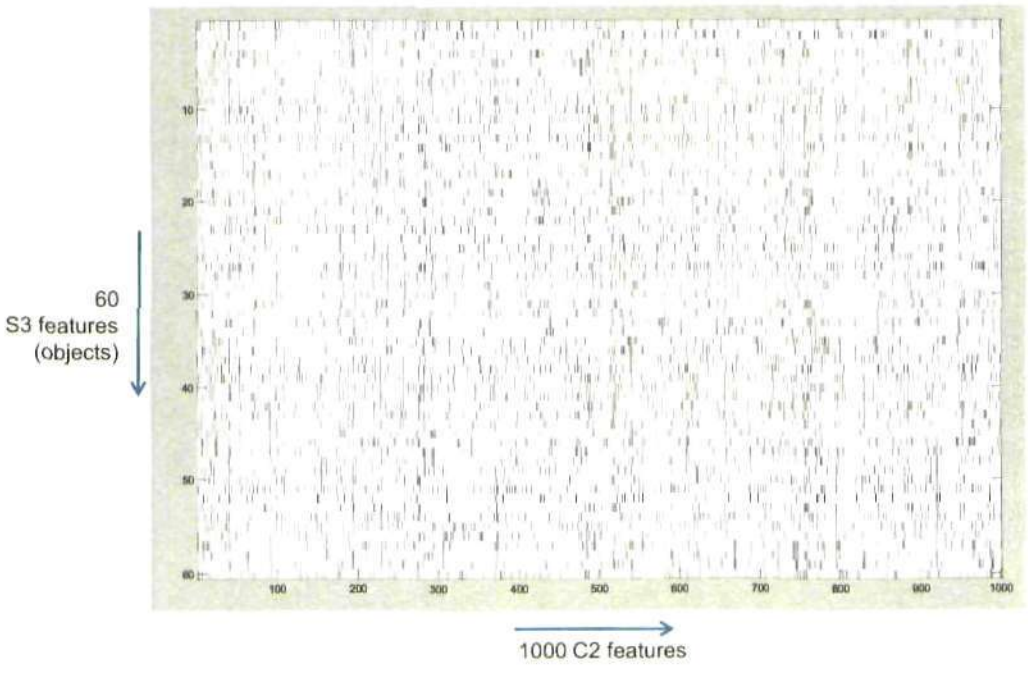


Figure 4.12: Weight matrix between C2 and S3 nodes for the 3-level architecture. The weights are learned in a supervised manner from the C2 response to each of the 60 training input images or objects. Each object is thus represented as a weighted subset of the 1000 C2 groups. The CPT $P(C2|S3)$ containing $K_{C2}(= 10000) \times K_{S3}(= 60)$ elements is derived from this weight matrix by using the same value for each of the $K_{C2group}$ features per group and sum-normalizing the weight matrices for each prototype.

240 prototypes or S3 states. This increases the invariance to position at the top level.

The learning process for the 4-layer architecture is not described in detail as it is a trivial extension of the methodology employed for the 3-layer architecture.

4.4 Feedforward processing

4.4.1 Approximation to the selectivity and invariance operations

For the feedforward recognition results presented in Chapter 4 we therefore assume that the network is a singly-connected tree, so that the λ messages can propagate to the root node without being affected by top-down messages (see Figure 3.9 in Section 3.3.3 for details). This is the same strategy used during the learning process. This facilitates the approximation to the HMAX operations and greatly reduces the computational costs to process each image. This is specially important when testing a large image dataset over a large parameter space.

Note that even if the root nodes and π messages were initialized to a flat distribution, they would still modulate the bottom-up λ messages, as π messages are multiplied by the CPT before being combined with the λ messages. In other words, as long as there is bottom-up evidence, it will be modulated by top-down messages even if the latter exhibit flat distributions. This was illustrated in Figure 3.7. Several recognition simulations were also performed without this assumption, in other words, with flat top-down messages that modulated the feedforward λ messages, in order to compare results and establish its validity. These revealed that a similar invariant recognition performance can be obtained even when including the feedback π messages (using loopy belief propagation), but performing a detailed systematic test over the complete dataset and parameter space is infeasible due to the high computational resources required.

The singly-connected tree assumption allows the selectivity operation in HMAX to be approximated as shown in Equation 4.9. Note that the original Radial Basis Function operation has been replaced by an approximately equivalent dot-product operation as proposed by Serre (2006), Serre et al. (2005a), and this dot-product operation is then approximated using the belief propagation equation. More precisely, the weighted sum over S1 locations and features is approximated as a sum over the features and a product over the locations. This can be interpreted as

the simultaneous coincidence of the features in the afferent nodes, as proposed by George and Hawkins (2009). Finally, the weight matrix for each S2 prototype is approximated by the CPT $P(C1|S2)$.

$$\begin{aligned} S2_{b_{S2},x_{S2},y_{S2},k_{S2}} &= exp\left(-\beta \cdot \left|\left|C1_{\{b_i,x_i,y_i\}} - P_{k_{S2}}\right|\right|^2\right) \\ &\approx \sum_{b_i,x_i,y_i} \sum_{k_i} C1_{\{b_i,x_i,y_i,k_i\}} \cdot W_{k_{S2}} \\ &\approx \prod_{b_i,x_i,y_i} \sum_{k_i} C1_{\{b_i,x_i,y_i,k_i\}} \cdot W_{k_{S2}} \\ &\cong \prod_{b_i,x_i,y_i} \sum_{k_i} \lambda(C1_{\{b_i,x_i,y_i\}} = k_i) \cdot P(C1_{\{b_i,x_i,y_i\}} = k_i | S2 = k_{S2}) \\ &= \lambda(S2_{b_{S2},x_{S2},y_{S2}} = k_{S2}) \end{aligned} \tag{4.9}$$

where the indices are given by Equations (4.3) to (4.5).

Similarly, the invariance operation in HMAX is approximated using belief propagation as shown in Equation 4.9. The approximation to the *max* operation is embedded in the $\lambda_{C1}(S2)$ output messages to S2 generated using the weights in the CPT $P(C1|S2)$, which sum over the C1 features of the same group. In order to make this possible, the most common S1 states and locations have previously been combined in the C1 node states through the CPT $P(S1|C1)$ (see Figures 4.7 and 4.9). In this sense it can be argued that both the selectivity operation and the invariance operation are actually implemented using the weights in $P(C1|S2)$, whereas the weights in $P(S1|C1)$ implement a necessary pre-processing step.

4.4.2 Dealing with large-scale Bayesian networks

Due to the large fan-in in the network and the large number of states, calculating the λ function of a node requires multiplying a high number of potentially very low probability values. For example, a C1 node in band 8 receives input from 968 (22×22 locations $\times$ 2 bands) S1 nodes, meaning that it is necessary to obtain the product of 968 probability distributions. The result

of this computation is often outside the typical numeric boundaries in simulation environments (for Matlab these boundaries range from 10^{-323} to 10^{+308}). For this reason it is necessary to make several approximations during the belief propagation calculation:

- Given a node X with child nodes $C_1, \cdots, C_M$, the number of input λ messages is reduced such that $\lambda(x) = \prod_{j \in \{j_{max}\}} \lambda_{C_j}(x)$, where $\{j_{max}\} \subset 1..M$, represents the indices of the M_{max} $\lambda_{C_j}(x)$ messages with highest variance, and $M_{max} \leq M$. The maximum number of input messages, M_{max}, is calculated as a function of the number of states of the messages, K_X, Matlab's maximum real value, $R_{max} = 10^{+308}$, and the minimium value allowed in probability distributions, V_{min}, as follows:

$$M_{max} = \frac{log(\frac{R_{max}}{K_X})}{log(\frac{0.1}{V_{min}})} \tag{4.10}$$

  Thus, the likelihood function of each node is obtained by multiplying only the M_{max} input λ messages with higher variance, where M_{max} is set to ensure that the result of the computation never reaches Matlab's numeric upperbound. Probability distributions with higher variance are chosen as they are likely to carry more information. In the majority of cases $M_{max} \geq M$, so the resulting computation is equivalent to the original belief propagation formulation.

  To check how well this sampling procedure managed to approximate the exact likelihood functions the method was tested statistically. Using randomly generated λ messages from a normal distribution, the difference between the exact likelihood and the approximated likelihood distribution obtained after sampling was measured, for different values of M_{max}. The difference was measured using the Kullback-Leibler (K-L) divergence which calculates the cross-correlation between an approximate distribution and the true distribution. This method cannot be considered a distance measure, as it is not symmetric, but has been used extensively to measure the goodness of fit between two discrete probability distributions (Friston and Kiebel 2009, Winn and Bishop 2005, Hinton et al. 2006).

Figure 4.13 shows the K-L divergence between the true and approximate likelihood distributions, averaged over 500 trials, as a function of M_{max} and the total number of parents N. The likelihood distributions are assumed to have $K = 100$ states. For comparison, the K-L divergence between the exact likelihood and a randomly generated likelihood distribution is also plotted.

The results show that the coefficient M_{max}/N increases as the goodness of fit between the approximation and the exact solution increases. Also, as the total number of input messages, N, increases, the goodness of fit decreases. The relative difference between the K-L divergence of the approximate and the random distributions suggests that for values of M_{max} above a given threshold the approximate distribution provides a good fit to the exact solution. It is important to note that in the real model data, the input λ messages are correlated (due to the overlap in receptive fields) and are therefore likely to present more similarities between them than the randomly generated λ messages of the statistical test. Additionally, a subset of the discarded distributions will typically present near-flat distributions as they originate from blank regions of the image. Consequently, the approximation in the model will constitute a better fit to the exact distribution than that suggested by this empirical test.

- The messages (probability distributions) are sum-normalized to 1 and then re-weighted so that the minimum value of the distribution is never below $V_{min} = 1/(10 \cdot K_X)$. All elements of the message that are below V_{min} are set to V_{min}. The overall increase in the sum of the elements of the resulting distribution is then compensated by proportionally decreasing the remaining elements (those which were not set the to V_{min}). Consequently, the resulting distribution will still be sum-normalized to 1, while having a minimum value equal to V_{min}. The distribution will have a profile equivalent to that of the original one, except for those elements that were originally below P_{min}, which will now exhibit higher relative values.

  This adjustment of the message probability distributions eliminates all values under V_{min}, thus allowing multiplicative combination of a greater number of input messages, i.e. M_{max}

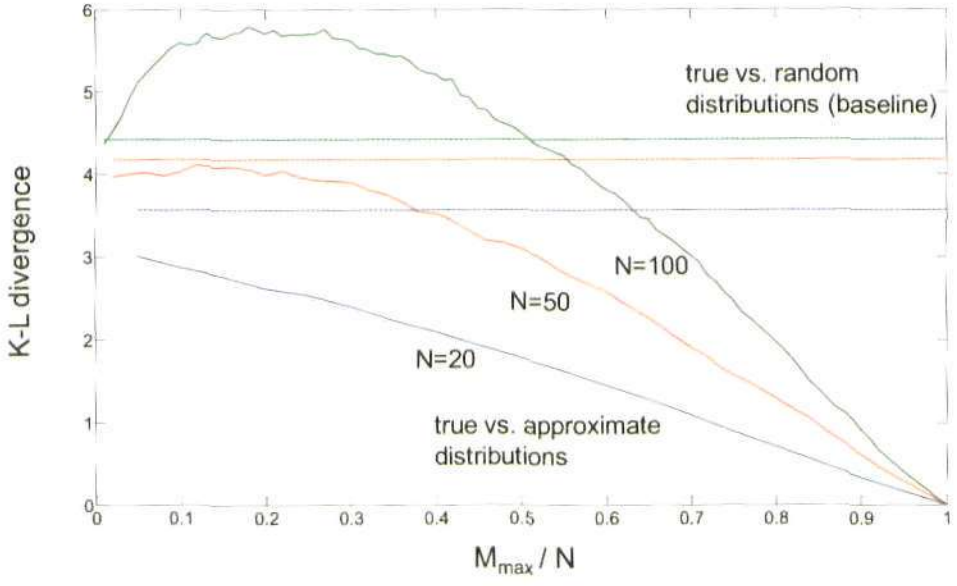


Figure 4.13: Kullback-Leibler divergence between the true and the approximate likelihood distribution for different values of M_{max}, averaged over 500 trials. The Kullback-Leibler (K-L) divergence, on the y-axis, measures the cross-correlation between an approximate distribution and the true distribution. This method cannot be considered a distance measure, as it is not symmetric, but has been used extensively to measure the goodness of fit between two discrete probability distributions (*Friston and Kiebel 2009*, *Winn and Bishop 2005*, *Hinton et al. 2006*). The x-axis shows the coefficient M_{max}/N, i.e. the percentage of λ messages of the total that are used in the approximation. Three different values of N are shown: 20 (blue lines), 50 (red lines) and 100 (green lines). The dotted horizontal line shows the K-L divergence between the true and a random distribution, which serves as a baseline to compare the goodness of fit of the approximate distribution.

is proportional to V_{min}.

4.5 Feedback processing

Section 4.4 describes how for the feedforward object recognition simulations the network was assumed to be singly-connected and tree-structured during the initial bottom-up propagation of evidence. This was done in order to simplify the computations and facilitate the approximation to the HMAX selectivity and invariance operations.

For the feedback simulations the network is not restricted by this assumption, and is thus allowed to maintain its multiply connected structure (multiple parents with loops). Evidence propagates simultaneously in both directions (up and down), at all layers of the network. The combination of parent messages is approximated using the weighted sum of *compatible parental configurations* method. To deal with loops, belief propagation becomes *loopy belief propagation*, which provides an approximation to the exact beliefs after several iterations. Further details of the feedback implementation and the approximations required due to the large dimensions of the network are included in this section.

4.5.1 Approximating π messages as beliefs

As shown in Equation 3.31, the outward π message generated at each node can be obtained as a function of its belief. The only difference is that the message from node X to C_j, i.e. $\pi_{C_j}(X)$ includes all incoming messages to X, except the one arriving from the destination node, i.e. $\lambda_{C_j}(X)$. This is done in order to avoid the circulation of duplicate information in the network.

However, for the purpose of simplification and increased computational performance, and only when the number of incoming messages is high, the outgoing $\pi_{C_j}(X)$ message can be approximated by the belief, $Bel(X)$. This approximation implies $\pi_{C_j}(X)$ also includes the evidence contained in $\lambda_{C_j}(X)$. However, $\pi_{C_j}(X)$ is calculated by combining messages from a total of $N+M$ nodes (all parent and children nodes), so the overall effect of one single message on the final message is proportional to $1/(N+M)$. This justifies the approximation in the present model where the values of N and M are in the order of hundreds or thousands. The same approximation is employed by other similar belief propagation models (Litvak and Ullman 2009,

George and Hawkins 2009).

4.5.2 Multiple parents

As described in Section 3.3.4 the number of elements of the CPT $P(X|U_1,\cdots,U_N)$ is exponential to the number of parents, N, as it includes entries for all possible combinations of the states in node X and its parent nodes, e.g. given $k_X = k_U = 4, N = 8$, the number of parameters in the CPT is $4 \cdot 4^8 = 262,144$. Additionally, the number of operations to compute the belief is also exponential to the number of parents, more precisely it requires k_u^N sums and $N \cdot k_u^N$ product operations. The exponential growth to the number of parameters and operations resulting from the combination of multiple parents is illustrated in Figure 4.14.

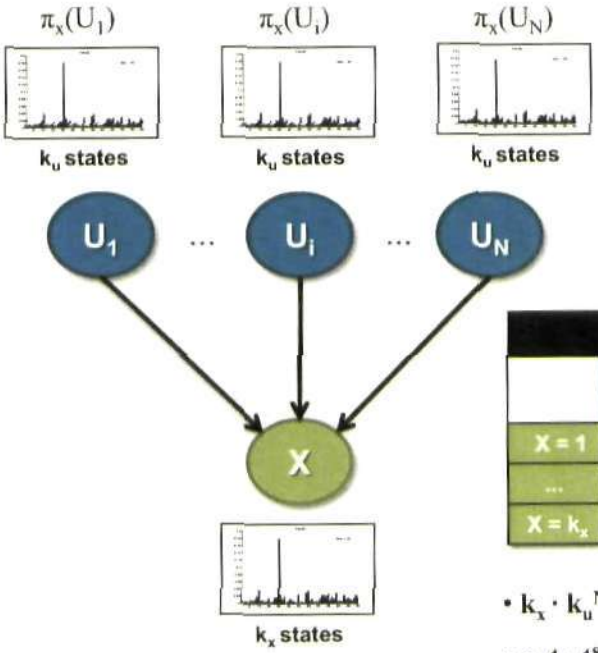
4.5.2.1 Weighted sum of compatible parental configurations

To solve the problem of the large number of entries in the CPT we implement the weighted sum of simpler CPTs based on the concept of *compatible parental configurations* (Das 2004) described in Section 3.3.4. This method obtains a $k_X \times k_U$ CPT, $P(X|U_i)$, between node X and each of its N parent nodes, and assumes the rest of the parents, U_j, where $j \neq i$, are in *compatible* states. The final CPT $P(X|U_1,\cdots,U_N)$ is obtained as a weighted sum of the N $P(X|U_i)$ CPTs. The total number of parameters required to be learned is therefore linear with the number of parents, more precisely, $k_X \cdot k_N \cdot N$. Using the values of the previous example, the number of elements is now $4 \cdot 4 \cdot 8 = 128$, several orders of magnitude smaller than the previous result. This is illustrated in Figure 4.15.

The *Learning* section (4.3) described 1) how to obtain the weight matrices between a parent node and its children, and 2) how to convert these weight matrices into individual CPTs for each of the child nodes. The resulting CPT is precisely in the form required to implement the weighted sum method, i.e. for each child node X there are N CPTs of the form $P(X|U_i)$, one for each of its parents. These can then be combined to form the final $P(X|U_1,\cdots,U_N)$.

4.5.2.2 Sampling from parent nodes

To reduce the excessive number of operations required to calculate the belief, only the k_{umax} states, with the highest values, from the N_{max} π messages, with the higher variance, are used



k_x = number of child states
k_u = number parent states
N = number of parent nodes

	$P(X \mid U_1,\ldots,U_N)$		
	$U_1,\ldots,U_N = \{1,\ldots,1\}$	...	$U_1,\ldots,U_N = \{k_u,\ldots,k_u\}$
X = 1			
...			
$X = k_x$			

- $\mathbf{k_x \cdot k_u^N}$ CPT parameters to learn **(exponential)**

e.g. $\mathbf{4 \cdot 4^8 = 262{,}144}$

- Belief calculation performs $\mathbf{k_u^N}$ *sums* and $\mathbf{N \cdot k_u^N}$ *product* operations **(exponential)**

$$Bel(X) = \prod_k \lambda_{C_k}(X) \sum_{U_1,\ldots,U_N} \left(P(X \mid U_1,\ldots,U_N) \prod_{i=1..N} \pi_X(U_i) \right)$$

Figure 4.14: Problems associated with the exponential dependency on the number of parent nodes. As described in Section 3.3.4, the number of elements of the CPT $P(X|U_1,\cdots,U_N)$ is exponential to the number of parents, N, as it includes entries for all possible combinations of the states in node X and its parent nodes, e.g. given $k_X = k_U = 4, N = 8$, the number of parameters in the CPT is $4 \cdot 4^8 = 262,144$. Additionally, the number of operations to compute the belief is also exponential to the number of parents, more precisely, it requires k_u^N sums and $N \cdot k_u^N$ product operations.

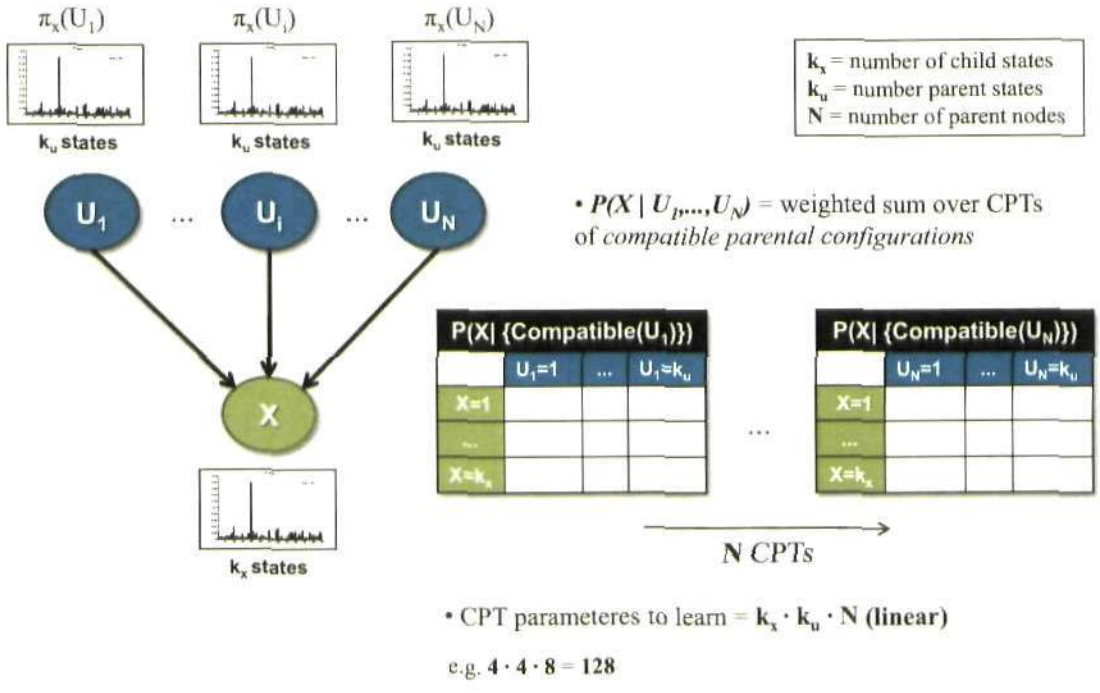
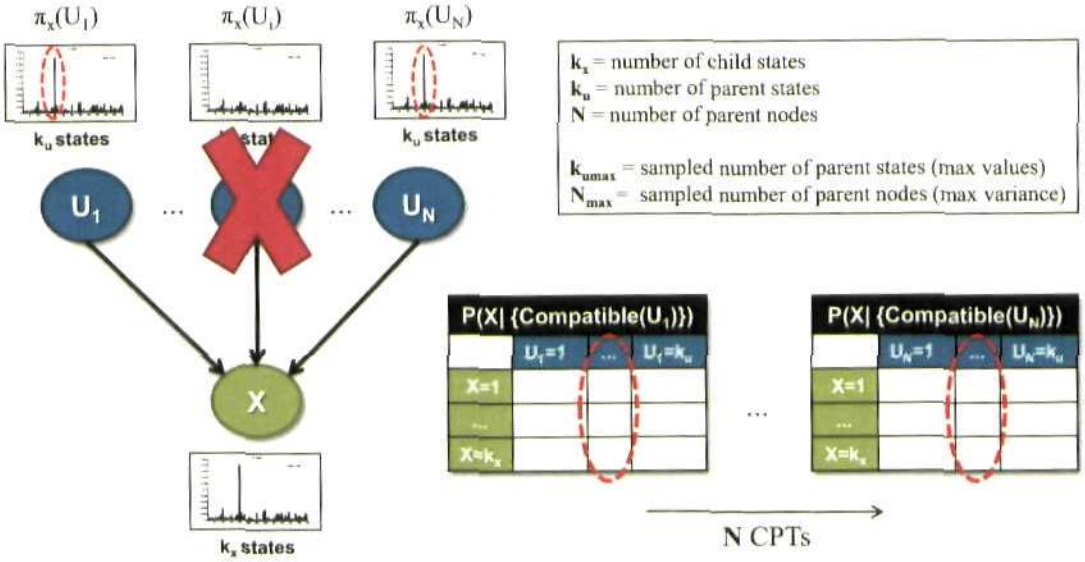


Figure 4.15: Approximation of the CPT between a node and its multiple parents using the weighted sum of N simpler CPTs (one per parent). This approach is based on the concept of *compatible parental configurations* (Das 2004) described in Section 3.3.4. The method obtains a $k_X \times k_U$ CPT, $P(X|U_i)$, between node X and each of its N parent nodes, and assumes the rest of the parents, U_j, where $j \neq i$, are in *compatible* states. The overall CPT $P(X|U_1, \cdots, U_N)$ is obtained as a weighted sum of the N $P(X|U_i)$ CPTs. The total number of parameters required to learn is no longer exponential, but linear with the number of parents, more precisely equal to $k_X \cdot k_N \cdot N$. Using the values of the example in Figure 4.14, the number of elements is now $4 \cdot 4 \cdot 8 = 128$, several orders of magnitude smaller than the previous result.



- Choose only the maximum $\mathbf{k_{umax}}$ values from the $\mathbf{N_{max}}$ parent nodes with highest variance
- Reduces the number of operations to calculate Belief and messages

Figure 4.16: Sampling of the parent π messages to reduce the number of operations required for belief propagation. Only the k_{umax} states, with the highest values, from the N_{max} π messages, with the highest variance, are used in the calculation, where $k_{umax} \leq k_u$ and $N_{max} \leq N$. The states with the stronger response of the probability distributions with higher variance are likely to carry most of the information content of the parent π messages. To ensure the belief calculations are still valid it is necessary to select the appropriate columns of the CPTs, i.e. those that correspond to the sampled states of the π messages. This reduces the number of operations to $k_{umax}^{N_{max}}$ sums and $N_{max} \cdot k_{umax}^{N_{max}}$ product operations.

in the calculation, where $k_{umax} \leq k_u$ and $N_{max} \leq N$. The states with the stronger response of the probability distributions with higher variance are likely to carry most of the information content of the parent π messages. To ensure the belief calculations are still valid it is necessary to select the appropriate columns of the CPTs, i.e. those that correspond to the sampled states of the π messages. This reduces the number of operations to $k_{umax}^{N_{max}}$ sums and $N_{max} \cdot k_{umax}^{N_{max}}$ product operations. Figure 4.16 illustrates the sampling process. Although in this section we refer only to the belief calculation, the same method is applied to calculate the λ messages, which also integrate information from the parent nodes.

To check how well this sampling procedure managed to approximate the exact belief functions, the method was tested statistically. Using randomly generated CPTs $P(X|U_1, \cdots, U_N)$ and like-

lihood functions $\lambda(x)$, the difference between the exact beliefs and the approximated beliefs obtained after sampling for different values of N_{max} and k_{umax} was measured. The difference was measured using the Kullback-Leibler (K-L) divergence.

Figure 4.17 shows the K-L divergence between the real and approximate beliefs, averaged over 50 trials, as a function of N_{max} and k_{umax}. For comparison, the K-L divergence between the exact belief and a randomly generated belief distribution is also plotted. The range over which these parameters are tested is limited by the computational cost associated with calculating the exact beliefs using CPTs of size exponential to the number of parents. Thus the chosen parameters are $k_X = 10, k_U = 20, N = 6, K_{umax} = \{1 \cdots 19\}$ and $N_{max} = \{1 \cdots 6\}$.

The results show that as N_{max} and k_{umax} increase, the goodness of fit between the approximation and the exact belief increases. Furthermore, the relative difference between the K-L divergence of the approximate and the random belief distributions suggests that even for relatively small values of N_{max} and k_{umax} the approximate belief provides a good fit to the exact belief. The sampling parameters have to be chosen as a compromise between the accuracy of the approximation and the computational cost.

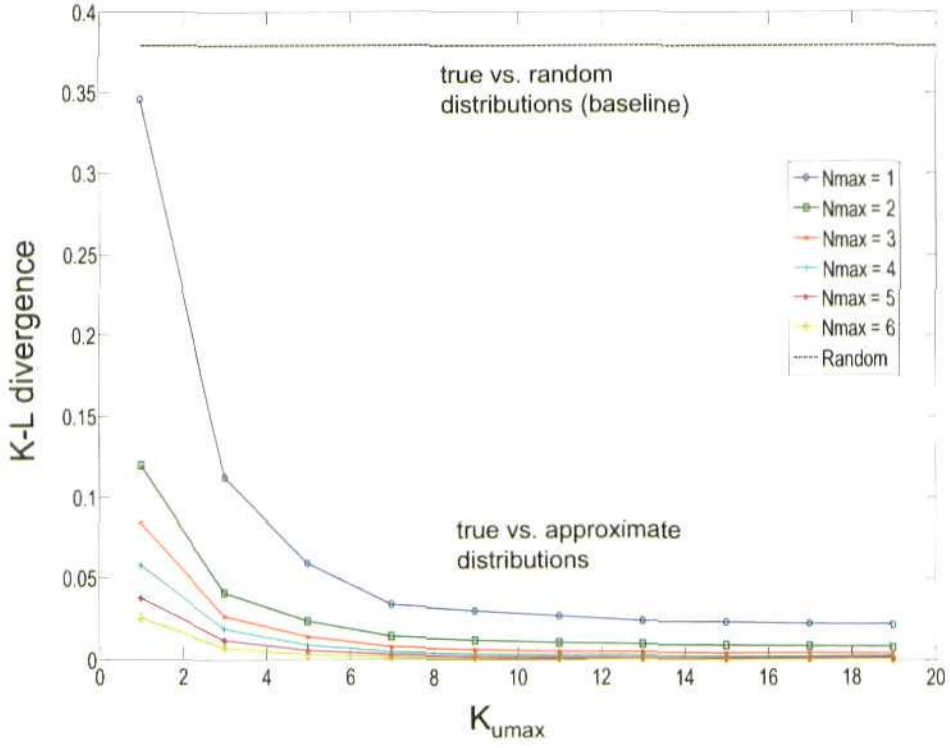


Figure 4.17: Kullback-Leibler divergence between the true and the approximate prior function $\pi(X)$ distribution for different values of k_{umax} and N_{max}, averaged over 50 trials. The Kullback-Leibler (K-L) divergence, on the y-axis, measures the cross-correlation between an approximate distribution and the true distribution, and is typically used as a goodness of fit between two discrete probability distributions (*Friston and Kiebel 2009, Winn and Bishop 2005, Hinton et al. 2006*). The x-axis shows the number of samples taken from the π messages, k_{max}. Results are plotted for values of N_{max} ranging from 1 to 6 as indicated in the colour legend. The dotted horizontal line shows the K-L divergence between the true and a random distribution, which serves as a baseline to compare the goodness of fit of the approximate distributions.

4.5.3 Loops in the network

4.5.3.1 Dynamic equations

Due to the overlap between the RF of nodes at all layers, the resulting Bayesian network has a large number of loops. As was described in Section 3.3.5, in Bayesian networks without loops belief propagation obtains the exact marginal probability distributions of all nodes after a set number of iterations. However, if the network has loops, the original belief propagation algorithm is no longer valid and approximate methods have to be implemented. The method selected for this model is loopy belief propagation, which has been empirically demonstrated to obtain good approximations to the exact beliefs in pyramidal networks (similar to that of the model) once the approximate beliefs have converged after several iterations (Weiss 2000).

The fact that belief propagation now requires several iterations means that a temporal dimension must be added to the original formulation. The resulting dynamical model is captured by the set of Equations 4.11. These also include the weighted sum method described in Section 3.3.4 to approximate the combination of top-down π messages.

$$
\begin{aligned}
Bel^{t+1}(x) &= \alpha \cdot \lambda^{t+1}(x) \cdot \pi^{t+1}(x) \\
\lambda^{t+1}(x) &= \prod_{j=1..M} \lambda^{t}_{C_j}(x) \\
\pi^{t+1}(x) &= \sum_{u_1,\ldots,u_N} P(x|u_1,\ldots,u_N) \cdot \prod_{i=1..N} \pi^{t}_{X}(u_i) \\
&= \sum_{u_1,\ldots,u_N} \sum_{g} w_g \cdot P(x|u_g) \cdot \prod_{i=1..N} \pi^{t}_{X}(u_i) \\
\lambda^{t+1}_{X}(u_i) &= \beta \sum_{x} \left[\lambda^{t+1}(x) \cdot \sum_{u_1,\ldots,u_N \setminus u_i} P(x|u_1,\ldots u_N) \cdot \prod_{k=1..N\setminus i} \pi^{t}_{X}(u_k) \right] \\
&= \beta \sum_{x} \left[\lambda^{t+1}(x) \cdot \sum_{u_1,\ldots,u_N \setminus u_i} \left(\sum_{g} w_g \cdot P(x|u_g) \right) \cdot \prod_{k=1..N\setminus i} \pi^{t}_{X}(u_k) \right] \\
\pi^{t+1}_{C_j}(x) &= \alpha \prod_{k=1..M\setminus j} \lambda^{t}_{C_k}(x) \cdot \pi^{t}(x) = \alpha \cdot \frac{Bel^{t+1}(x)}{\lambda^{t}_{C_j}(x)} \approx Bel^{t+1}(x)
\end{aligned}
\tag{4.11}
$$

4.5.3.2 Dynamics of loopy belief propagation

Loopy belief propagation also requires all λ and π messages to be initialized to a flat distribution, so that, even during the first iteration, all nodes of the network propagate upward and downward messages. Except for the λ messages from the dummy nodes, which will contain evidence from the image, and the π messages from the root nodes, which will propagate the *prior*, the rest of the messages will propagate flat distributions during the first time step. During the following time steps the dummy nodes' evidence will propagate to the top layers, merging with the downward prior information and being modulated at each layer by the inherent parameters of the network contained in the CPTs. The dynamics of loopy belief propagation in the proposed Bayesian network are illustrated in Figure 4.18.

The computational cost of updating the beliefs of nodes at all layers at every time step is very high. An alternative approach to reduce this cost is to update only the belief of a given layer at each time step as in tree-structured networks. For the majority of results present in this thesis the model implemented an *upward* belief update as opposed to the *complete* belief update. This is illustrated in Figure 4.19. For purposes of clarity each simulation step, t_{sim}, thus consists of five loopy belief propagation steps. This simplification of loopy belief propagation is justified in the sense that evidence arrives from the lower layer dummy nodes and thus only the belief of the nodes in the adjacent layer will provide meaningful information. All the computation required to calculate the π messages and beliefs in the upper layers during the first time-steps shown in Figure 4.18 is now saved. Further, it means evidence propagated from the dummy nodes will only be modulated at each layer by the initial flat top-down π messages, thus increasing the chance of a good recognition performance. The main disadvantage of this method is the asymmetry between bottom-up and top-down propagation of evidence, as a belief update or

[4]Caption for Figure 4.18. Dynamics of loopy belief propagation in the proposed model. At t=0 all messages are initialized to a flat distribution (symbolized with a *1* in the figure) except for the λ message from the dummy nodes and the top level π message or prior distribution. At t=1, once the initial flat π messages are multiplied by the corresponding node CPTs they generate non-flat beliefs and subsequent non-flat π and λ messages (see Figure 3.7 for a numeric example). The non-flat feedforward λ messages, λ_{dummy}, will also modulate the belief at each node and subsequent λ messages generated. However, the λ message generated by nodes with an incoming flat λ message will also generate flat output λ messages. For this reason, it takes 4 time-steps (the diameter of the network) to propagate the lower level evidence, λ_{dummy}, to the top node. The bottom-right image symbolically illustrates the existence of loops in the network and how this leads to the recursive nature and double-counting of messages in loopy belief propagation.

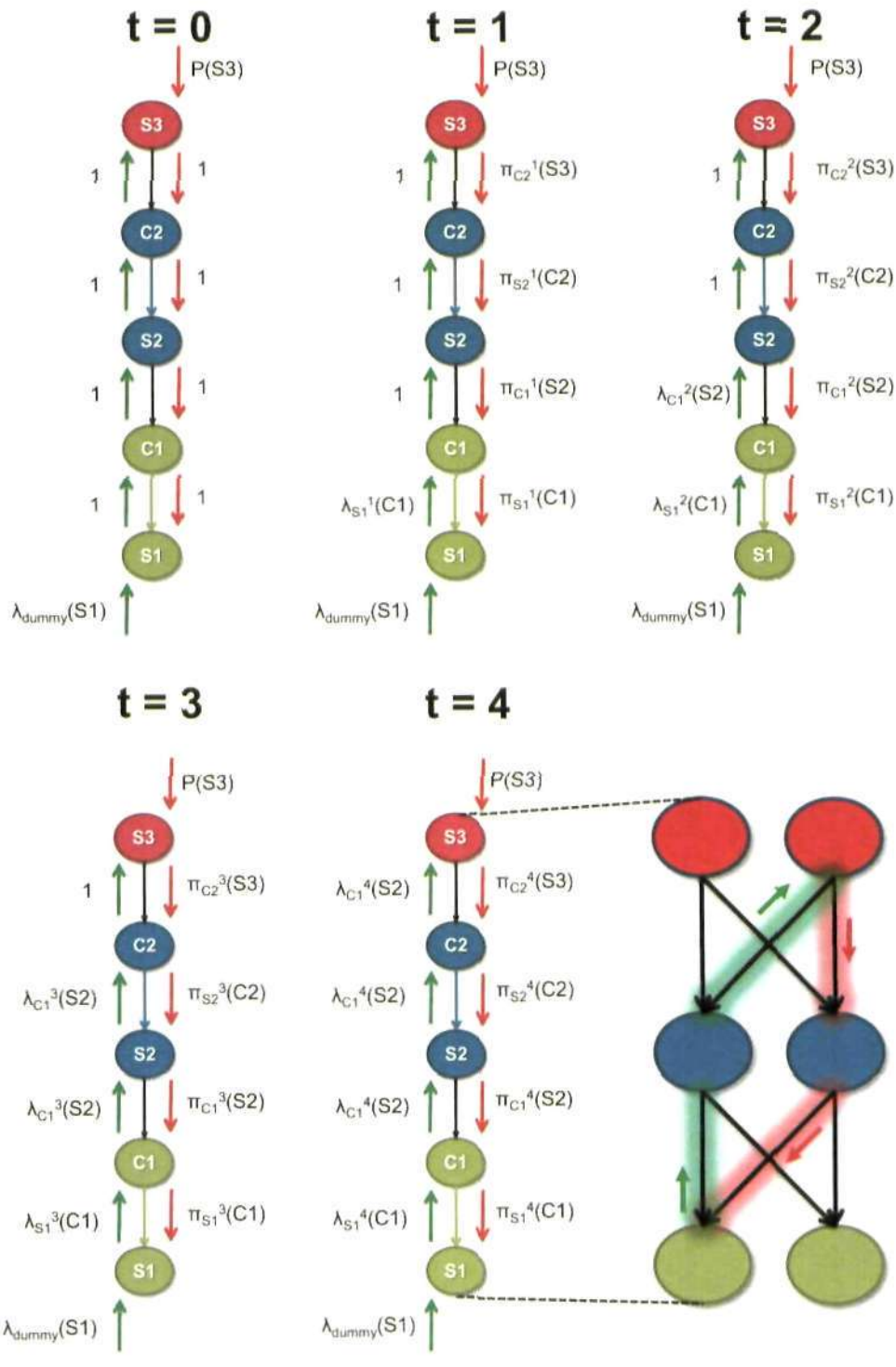


Figure 4.18: For caption see footnote[4].

evidence in the top layer will take four simulation time steps, t_{sim}, to reach S1, whereas evidence from S1 reaches the S3 layer in one simulation time step.

A third belief update method can be implemented, namely the *up-down* belief update method shown at the bottom of Figure 4.19. This method propagates bottom-up evidence and top-down evidence at the same rate, such that a single simulation time step, t_{sim}, will update all beliefs in an up-down pass. However, it means that intermediate nodes are updated more often than peripheral nodes and the longer simulation time steps makes it more difficult to analyze the results. For this reason, the *upward* belief update method was implemented in most of the model simulations. However, for comparison, an example of the results obtained using the *complete* and *up-down* belief update methods is provided. Although ideally the *complete* belief update method should be implemented, both the *upward* and *up-down* methods provide interesting and less computationally demanding alternatives, which, nonetheless, have corresponding disadvantages.

4.5.3.3 Accumulation of belief responses

As stated before, the λ message generated by nodes with an incoming flat λ message will also generate flat output λ messages, even though the belief of the node might be non-flat due to the π messages (for an example see Figure 3.8). This is the reason why the λ messages from nodes in Figure 4.18 show flat distributions until the evidence from the dummy node arrives. However, this also implies an important limitation as it means belief responses do not accumulate over time. In other words, regions with incoming flat λ messages (e.g. missing contours in Kanizsa figure) will generate flat outgoing λ messages even if the belief of the node shows a non-flat distribution (e.g. illusory contour). To overcome this problem, for some of the results presented in Chapter 5, the λ message equation was modified to be based on the current belief of the

[5]Caption for Figure 4.19. Comparison of three different belief update methods during loopy belief propagation. *Top*) The complete belief update method, which is the most rigourous approach to obtain a good approximation of the beliefs, but is very computationally expensive. *Middle* The upwards belief update method, which updates the belief of the nodes in one layer at every time step, starting from the bottom layer where the evidence originates. The main disadvantage is that evidence from S3 takes four t_{sim} to reach S1, while evidence from S1 takes 1 t_{sim} to reach S3. This method was employed to obtain most of the results in the thesis. For clarity, a simulation time step, t_{sim}, is made equal to 5 loopy belief propagation time steps. *Bottom*) The up-down belief update method which manages to propagate bottom-up and top-down evidence at the same rate, although intermediate layers are updated more often than peripheral layers. See main text for a more detailed comparison between the methods.

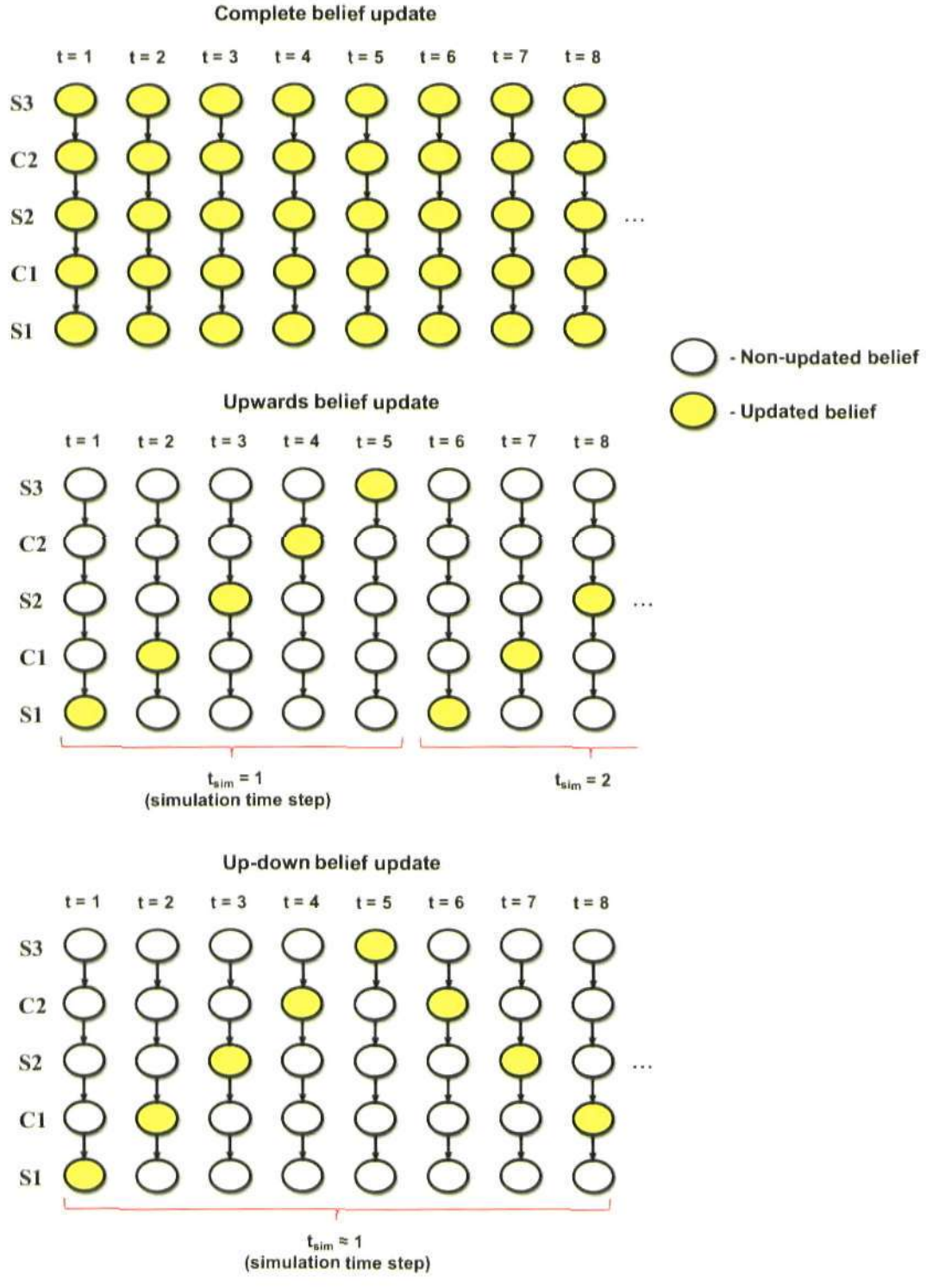


Figure 4.19: For caption see footnote[5].

node instead of on the likelihood function, $\lambda(X)$. This allows the λ messages to be modulated by top-down feedback even in regions that have initially flat distributions, such as the missing contours in the Kanizsa figure. This modification is shown in Equation (4.12).

$$\lambda_X^{t+1}(u_i) = \beta \sum_x \left[Bel^{t+1}(x) \cdot \sum_{u_1,\ldots,u_N \setminus u_i} \left(\sum_g w_g \cdot P(x|u_g) \right) \cdot \prod_{k=1..N \setminus i} \pi_X^t(u_k) \right] \tag{4.12}$$

4.6 Summary of model approximations to Bayesian belief propagation

- Feedforward recognition results assume a singly-connected tree-structured network (no loops and one parent per node) so the HMAX operations can be approximated by the propagation of bottom-up λ messages. Similar approaches have been used in other related models (Epshtein et al. 2008, George and Hawkins 2009, Hinton et al. 2006). Preliminary results suggest a similar invariant recognition performance can be obtained even when including the feedback π messages (loopy belief propagation), but the computational cost associated precludes a comprehensive systematic test over the complete dataset and parameter space.

- The number of input $\lambda_{C_i}(x)$ messages used to compute the likelihood function $\lambda(x)$ is limited to M_{max}, in order to prevent the result of the product operation from being outside of Matlab's numeric range. The method has been empirically demonstrated to provide a relatively good fit to the exact distribution given a moderate value of M_{max}.

- The λ and π messages are sum-normalized to 1 and then re-weighted so that the minimum value of the distribution is equal $V_{min} = 1/(K/10)$. This prevents extremely low values leading to out of range solutions during the belief propagation operations. The overall shape of the distribution remains identical, except for some of the elements with smaller values, which may now exhibit a relatively larger value. However, the states with lowest values are less likely to affect the final result in a significant way and many of them will be discarded anyway during the sampling methods implemented.

- The π messages are approximated by the belief at each node. The same approach is

used by Litvak and Ullman (2009), George and Hawkins (2009). This approximation is justified when the total number of incoming messages to a node is relatively high, as is the case of the present model

- The CPT $P(X|U_1,\cdots,U_N)$ is approximated as the weighted sum of N CPTs of the form $P(X|U_i)$. The method has been justified geometrically as providing a good model of the combination of information from multiple experts (parent nodes) and has been successfully employed on other probabilistic models that require reasoning under uncertainty (Das 2004).

- For the calculation of the belief and the λ messages, only k_{umax} highest-valued samples from the N_{max} π messages with the highest variance are employed. The method has been empirically demonstrated to provide a relatively good fit to the exact distribution, given moderate values of N_{max} and k_{umax}.

- To avoid the excessive computational cost associated to updating the beliefs and output messages of the nodes in all layers, beliefs are for a single layer at each time step, starting from the bottom layer and moving upwards sequentially. The rationale behind this approximation to loopy belief propagation is that evidence arrives at the network from dummy nodes connected to the bottom layer.

4.7 Original contributions in this chapter

- A Bayesian network that captures the structure of the HMAX model, a hierarchical object recognition model based on anatomical and physiological cortical data.

- An approximation to the selectivity and invariance operations of the HMAX model using the belief propagation algorithm over the proposed Bayesian network.

- An inherent extension of the static feedforward HMAX model to include dynamic and recursive feedback based on the loopy belief propagation algorithm.

- A particularization of the CPT learning method proposed by Das (2004) to the hierarchical object recognition domain. The method simplifies the generation of the CPT parame-

ters for Bayesian networks where nodes have multiple parents.

- Solutions to the problems associated with the integration of information in large-scale Bayesian networks. These include sampling methods and the re-weighting of probability distributions to establish a minimum value.

Chapter 5

Results

5.1 Feedforward processing

5.1.1 Layer by layer response

This section provides details of the internal representation of the image maintained by each of the layers. It serves to illustrate how the probabilistic representation compares with the classical HMAX model representation and facilitates understanding of the feedback results. All the results in this section are for the first input image of the training dataset, namely the letter A.

Figure 5.1 shows the response of the battery of Gabor filters at four different orientations and 16 different sizes (bands) applied to the input image. The bottom-up messages from the layer of dummy nodes, $\lambda_{dummy}(S1)$, are made up of the responses to the four orientations at each location and scale band.

Figure 5.2 shows the likelihood response of the S1 nodes, $\lambda(S1)$, obtained by sum-normalizing the input $\lambda_{dummy}(S1)$ messages. The grey colour over the blank input regions indicates that all four orientations are equally probable at that location, thus each one has a value of $1/4$.

Figure 5.3 shows the likelihood response of the C1 nodes, $\lambda(C1)$. The 2D maps represent the sum of the features in each C1 group at each location. Figure 5.4 shows the response of the C1 units in the original HMAX model, in other words, calculating the *max* over the S1 afferent units, for the same parameter set. This allows one to compare the response between the C1 nodes in the model proposed and the C1 nodes in the original HMAX model.

Figure 5.5 shows the likelihood response of the S2 nodes, $\lambda(S2)$, at all locations of a specific band and RF size. The figure also includes a reconstruction of the S2 internal representation

using the C1 features of the maximum S2 prototypes at each location. This graphical view of the S2 internal representation is limited in that it is based only on one of the 250 values of each distribution.

Figure 5.6 shows the likelihood response of the C2 nodes, $\lambda(C2)$, at all locations and RF sizes. The figure also includes a reconstruction of the C2 internal representation using the C1 features corresponding to the maximum C2 prototype at each location. This graphical view of the C2 internal representation is limited in that it is based only on one of the 2500 values of each distribution. Also, due to the large size and great overlap between the RF of the C2 nodes, the maximum feature is likely to be the same for adjacent nodes. Note the C2 response is shown for the alternative 3-level architecture (based on Yamane et al. (2006)), such that different architectures will exhibit different number of nodes, for instance the 3-level architecture has only one C2 node.

Figure 5.7 shows the likelihood response of the S3 nodes, $\lambda(S3)$, at all locations and RF sizes of the alternative 3-layer architecture. In this case the input image used was the 24th object of the dataset, which corresponds to S3 states 93 to 96 of the 240. To recap, in this architecture there are four S3 prototypes for each object, corresponding to four different possible locations of the object at the C2 level. Beside the maximum value of each distribution is shown an image which symbolically corresponds to the S3 prototype of that S3 state. Note that, in most cases, the winner element corresponds to a prototype of the input image, although in some cases there might exist some ambiguity and other similar object prototypes may exhibit relatively large values.

For further details see the figure captions.

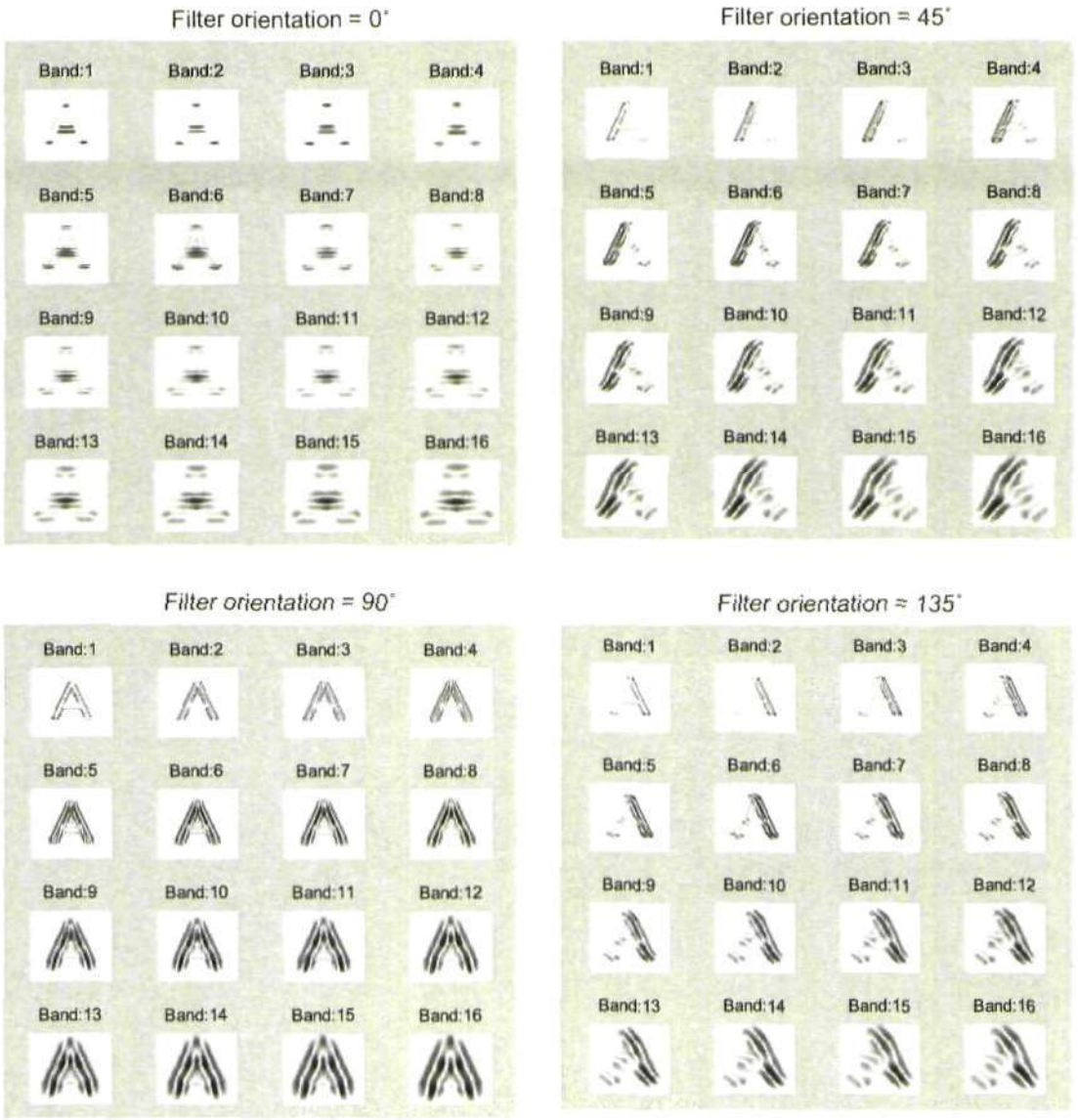


Figure 5.1: Response of the Gabor filters used to generate the $\lambda_{dummy}(S1)$ messages. The input image (the letter *A*) is filtered by Gabor filters at four different orientations ($0^{\circ}, 45^{circ}, 90, 135^{circ}$ and 16 different sizes or scale bands (ranging from 7 pixels to 37 pixels in steps of two pixels). The $\lambda_{dummy}(S1)$ messages are made up from the responses to the four orientations at each location and scale band.

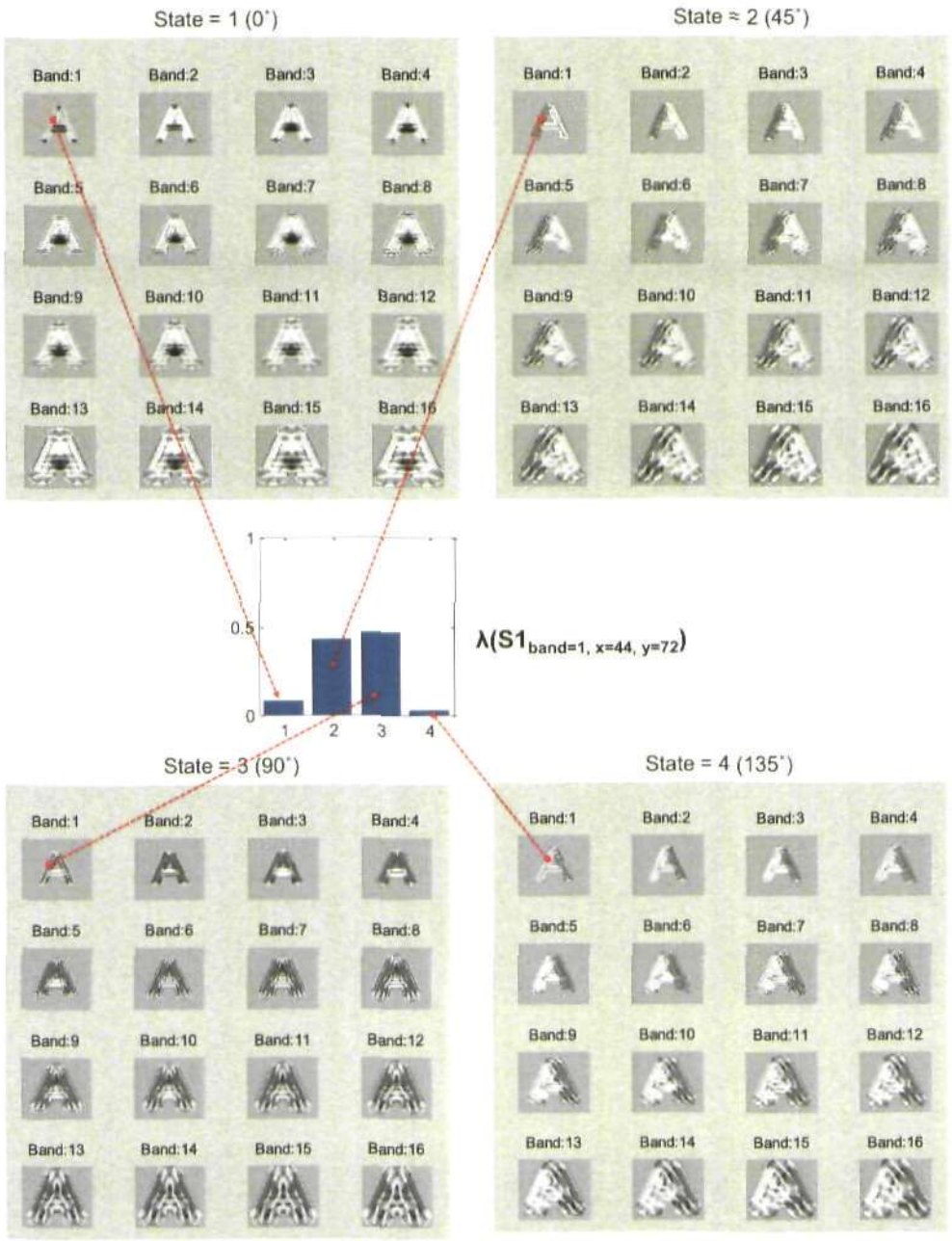


Figure 5.2: Likelihood response of the S1 nodes, $\lambda(S1)$. Responses are shown as a 2D map over the locations of the nodes for each state. The $\lambda(S1)$ of a specific node is shown in the centre of the image, the red dotted arrows indicate from where in the 2D maps the values come. The $\lambda(S1)$ distribution is obtained by sum-normalizing the input $\lambda_{dummy}(S1)$ messages. The grey colour over the blank input regions indicates that all four orientations are equally probable at that location, thus each one has a value of $1/4$.

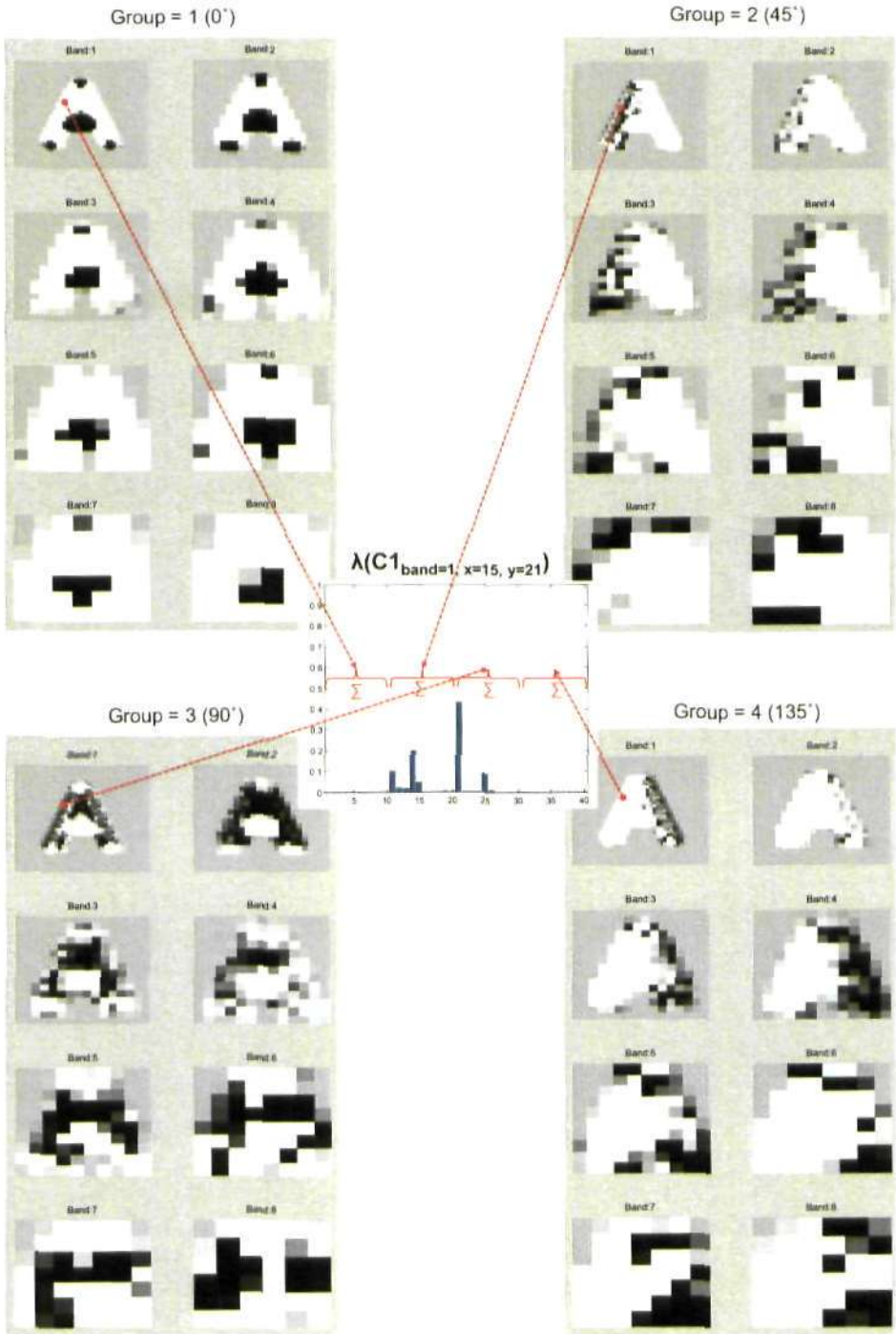


Figure 5.3: Likelihood response of the C1 nodes, $\lambda(C1)$. Responses are shown as a 2D map over the locations of the nodes for each group of states. Each C1 group corresponds to one of the S1 states or filter orientations. The $\lambda(C1)$ of a specific node is shown in the centre of the image, the red dotted arrows indicate from where in the 2D maps the values come from.

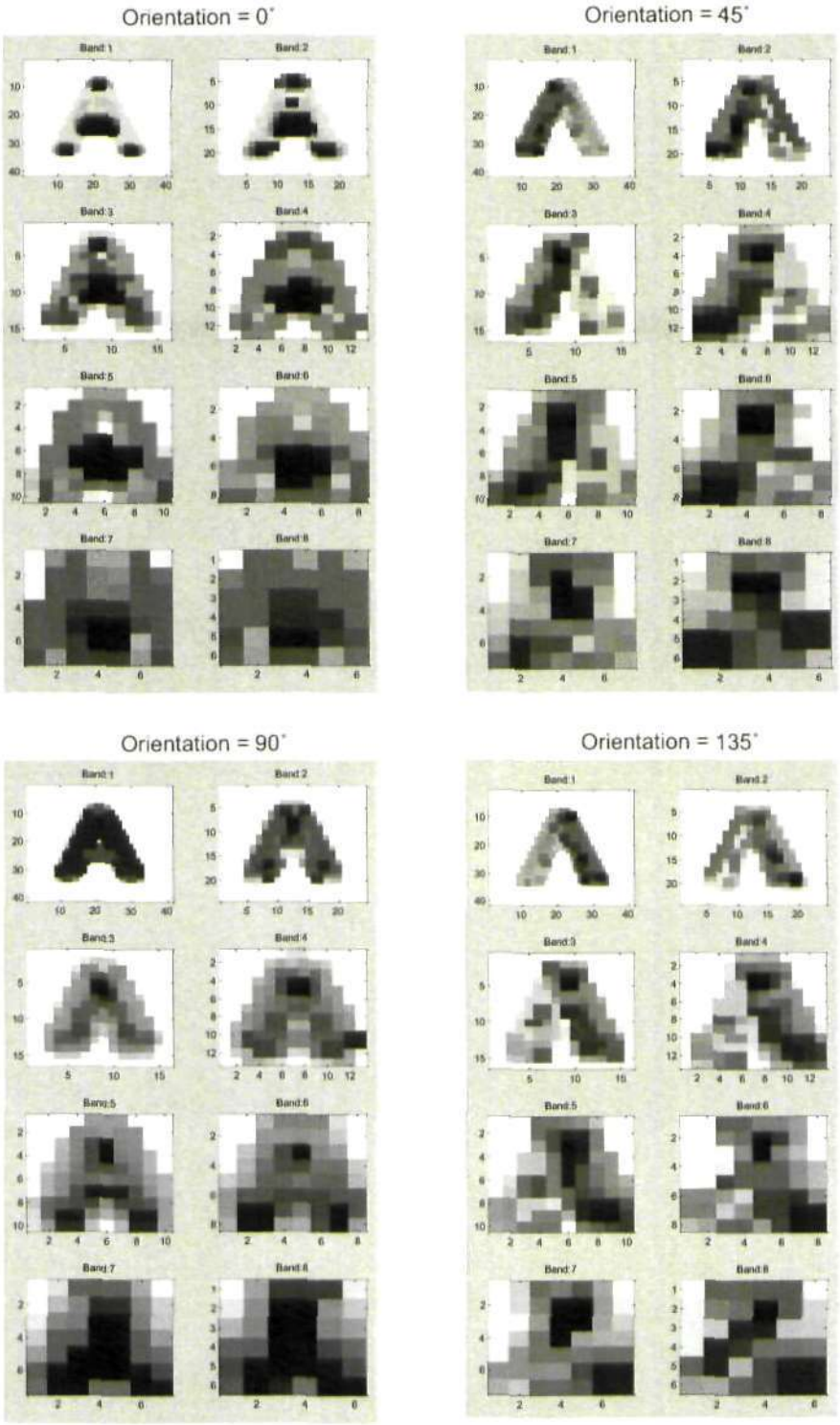


Figure 5.4: Response of the C1 units in the original HMAX model. The response is calculated as the *max* over the S1 afferent units. The input image and parameters are equivalent to those used to obtain the $\lambda(C1)$ in Figure 5.3. This enables the comparison between the response of the C1 nodes in the model proposed and the C1 nodes in the original HMAX model.

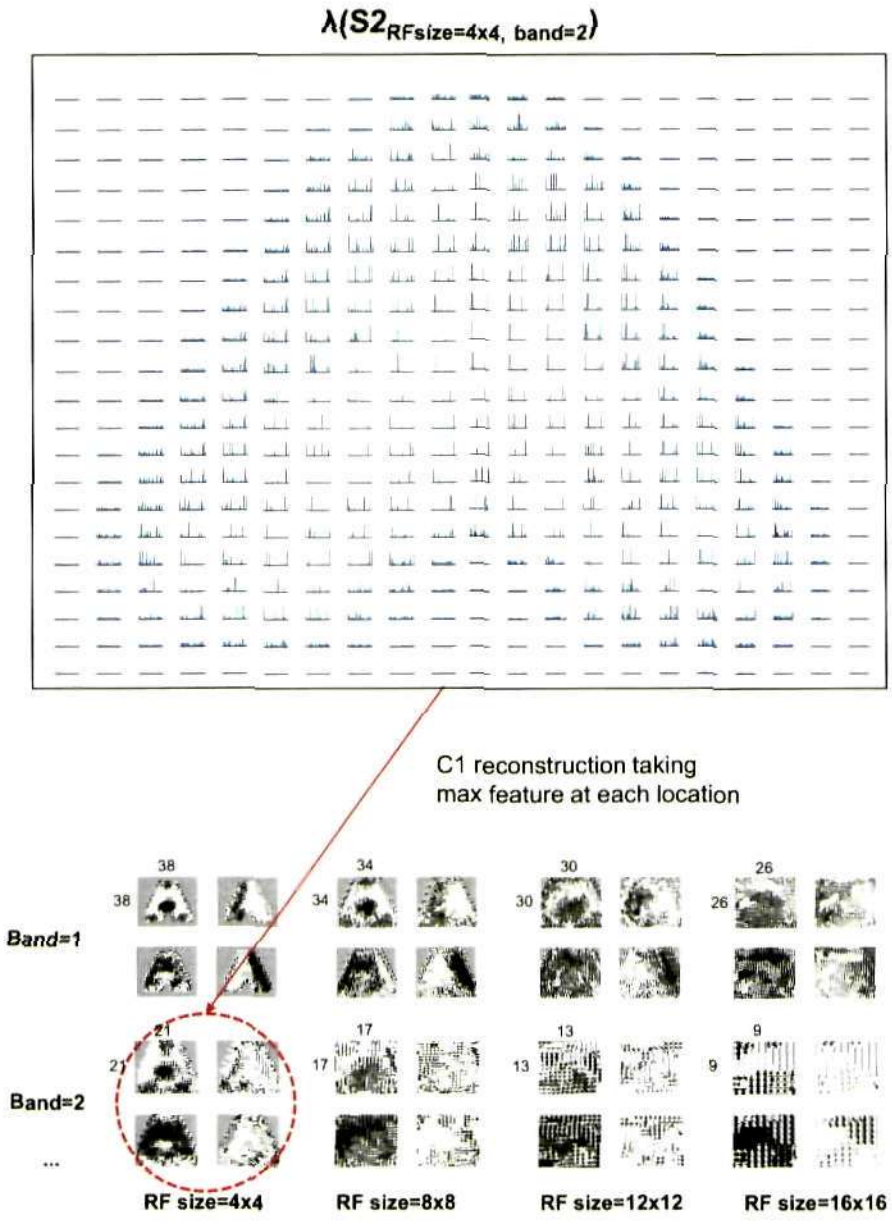


Figure 5.5: Likelihood response of the S2 nodes, $\lambda(S2)$. *Top)* The $\lambda(S2)$ probability distributions for the S2 nodes with RF size=4x4 ($N_{S2} = 4$) and band=2 at all the 21x21 spatial locations. *Bottom)* Reconstruction of the S2 internal representation using the C1 features that make up each S2 prototype (S2 feature). At each location the S2 prototype (a group of spatially arranged C1 features) corresponding to the max S2 feature is shown. This provides a graphical view of the S2 internal representation. Note that there is much more information contained in all the other values of the distribution, which is not represented graphically.

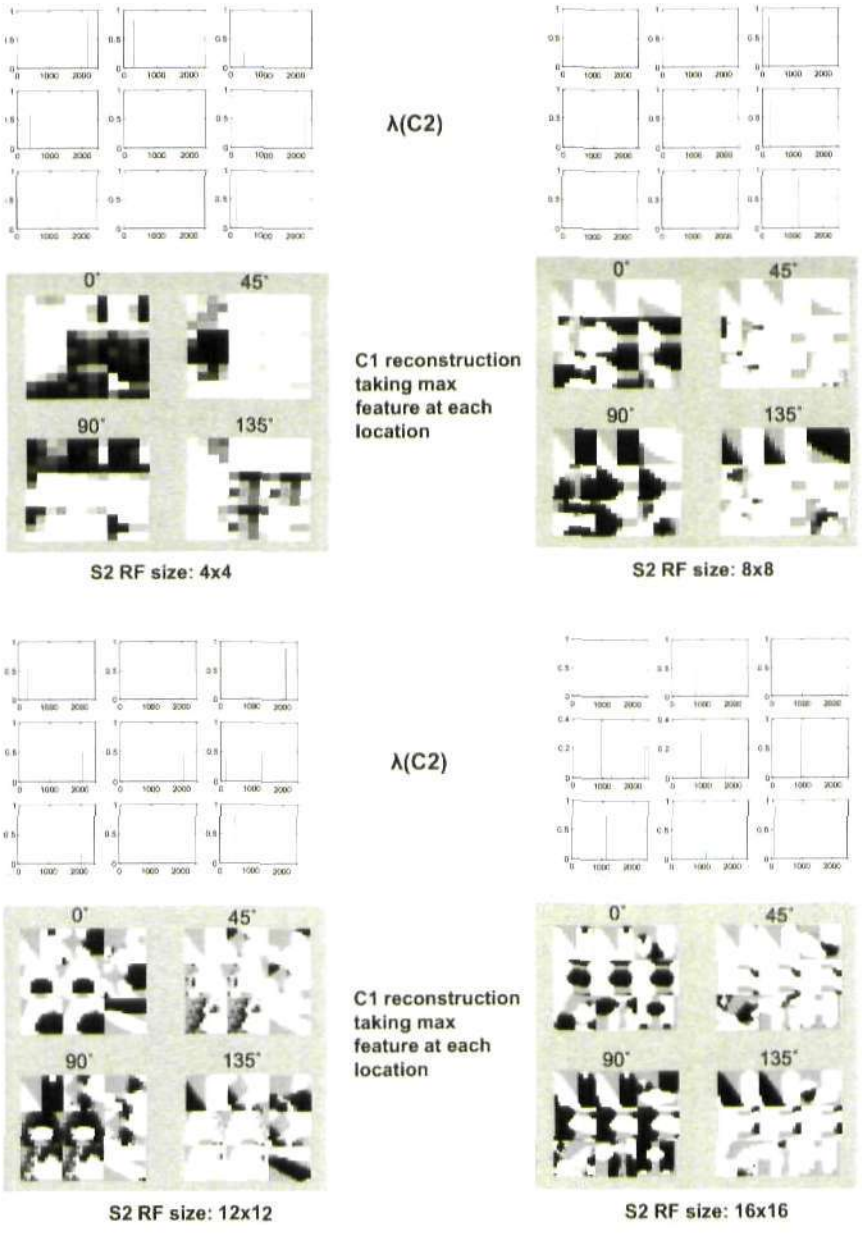


Figure 5.6: Likelihood response of the C2 nodes, $\lambda(C2)$, at all locations and S2 RF sizes. Below each distribution is shown the reconstruction of the C2 internal representation using the C1 features corresponding to the maximum C2 prototype at each location. This graphical view of the C2 internal representation is limited in that it is based only on one of the 2500 values of each distribution. Due to the large size and great overlap of the RF of the C2 nodes, the maximum feature is likely to be the same for adjacent nodes. The C2 response shown here is for the alternative 3-level architecture (based on Yamane et al. (2006)).

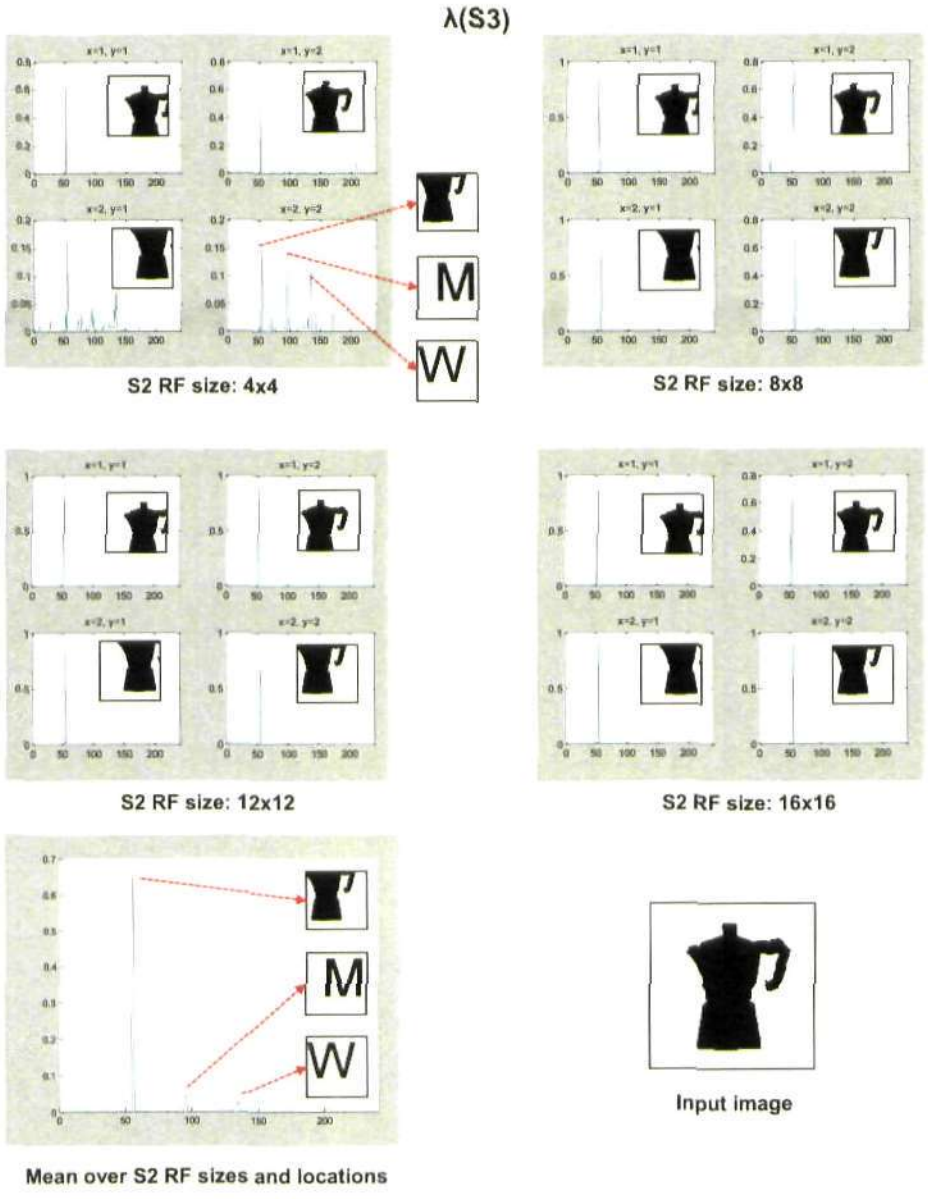


Figure 5.7: Likelihood response of the S3 nodes, $\lambda(S3)$, at all locations and RF sizes of the alternative 3-layer architecture. The input image used was the 24th object of the dataset, which corresponds to S3 states 93 to 96 of the 240 states. Recall in this architecture there are four S3 prototypes for each object, corresponding to four different possible locations of the object at the C2 level. An image next to the maximum value of each distribution is shown to symbolically represent the winner S3 prototype. Note that in most cases the winner element corresponds to a prototype of the input image. However, in some cases there might exist some ambiguity and other similar object prototypes may exhibit relatively large values. For example, for location (2,2) of RF size=4x4, the prototypes of the letters *M* and *W* also showed relatively high values. The bottom-left distribution shows the mean S3 likelihood response over the 2x2 locations and the four RF sizes.

5.1.2 Object categorization

This section describes the performance of the model during feedforward categorization, based on the feedforward processing constraints defined in Section 4.4. The network was trained using 60 object silhouette images, shown in Figure 5.8, from which the S2 and S3 prototypes were learned. The trained network was then tested on different transformations of the same images including occluded, translated and scaled versions.

Throughout this section, I have used *correct categorization* to mean that the state with the maximum value of the model's top layer response corresponds to the input image. For the 3-level and 4-level architectures, the distributions of the four top layer nodes, corresponding to each of the four S2 RF sizes, are averaged, resulting in a single distribution with 60 states. Additionally, for the alternative 3-level architecture, the values of the four prototypes learned for each object category are also averaged, leading again to a single distribution with 60 states. The model's performance is measured as a percentage of correctly categorized images for each dataset of 60 images.

For the occluded test set an average of 30% of the image's black pixels are deleted using a rectangular white patch. The rectangle is placed in a position that leaves the image identifiable to a human observer. In the translated test-set, the object is moved to a new position within the available image frame of 160x160 pixels. The displacement will be near to the maximum permitted in both directions but will depend on the original object size, i.e. small objects allow for bigger displacements. Two different scale sets have been used: scale $\pm 10\%$, where the image is scaled to either 90% or 110% of the original size and centred; and scale $\pm 20\%$, where the image is scaled to either 80% or 120% of the original size and centred. An example of the different transformations for five arbitrary images is shown in Figure 5.9.

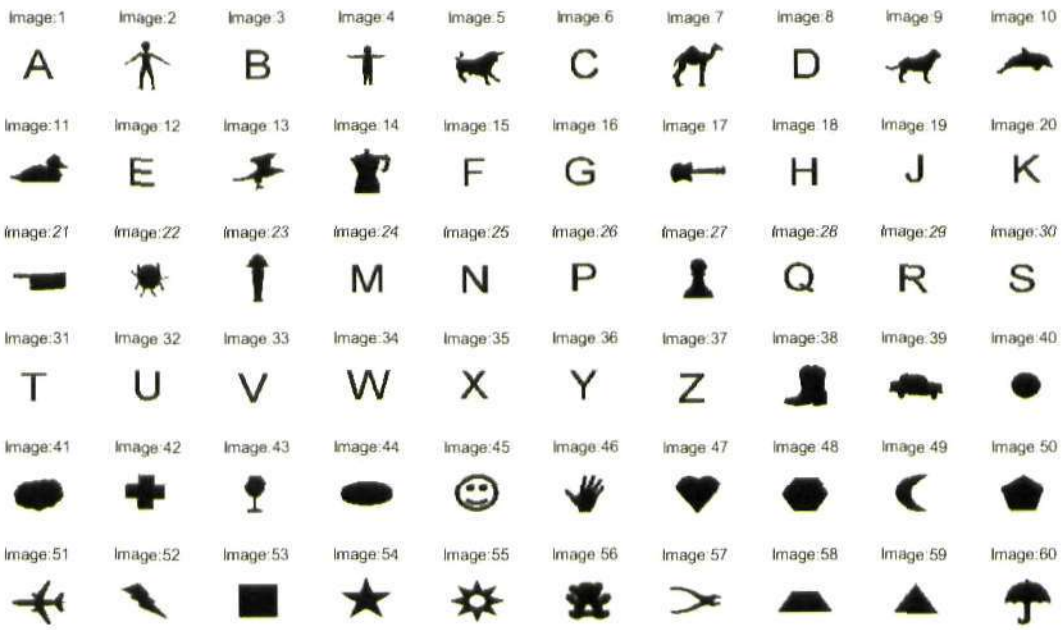


Figure 5.8: Dataset of 60 object silhouette images used to train and test the model. The S2 and S3 prototypes were learned from this set of images. The trained network was then tested on different transformations of the same images including occluded, translated and scaled versions.

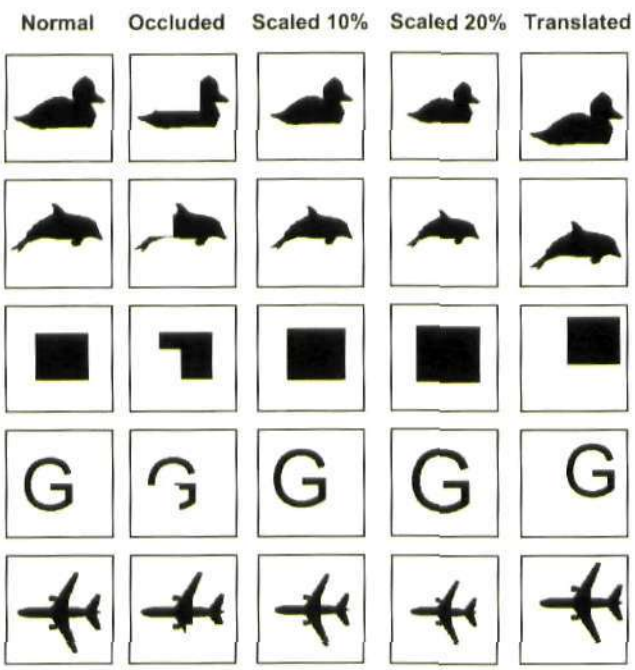


Figure 5.9: Examples of object transformations. The trained network was then tested on different transformations of the training images including occluded, translated and scaled versions. Examples of these transformations are shown here for five objects.

5.1.2.1 Categorization as a function of the number of states per group in complex layers

Figure 5.10 shows the categorization performance as a function of the number of states per group in the C1 layer, $K_{C1group}$, while Figure 5.11 shows the categorization performance as a function of the number of states per group in the C2 layer, $K_{C2group}$. Results were obtained for the 3-level architecture and are plotted for the five test datasets as detailed in the figure legend.

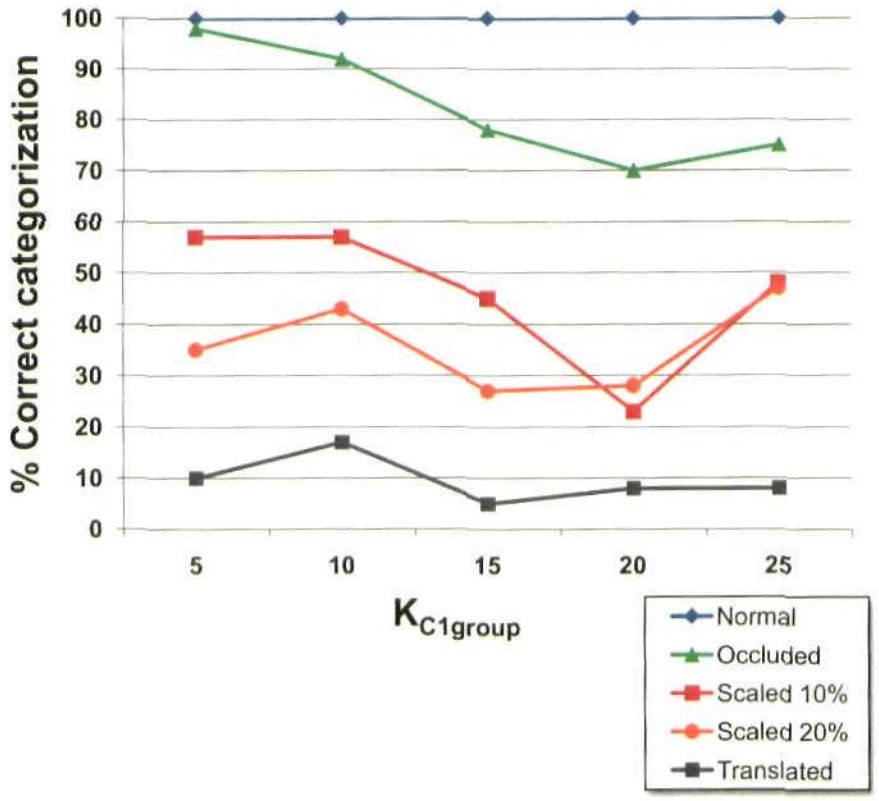


Figure 5.10: Categorization performance as a function of the number of states per group in the C1 layer, $K_{C1group}$, for the 3-level architecture.

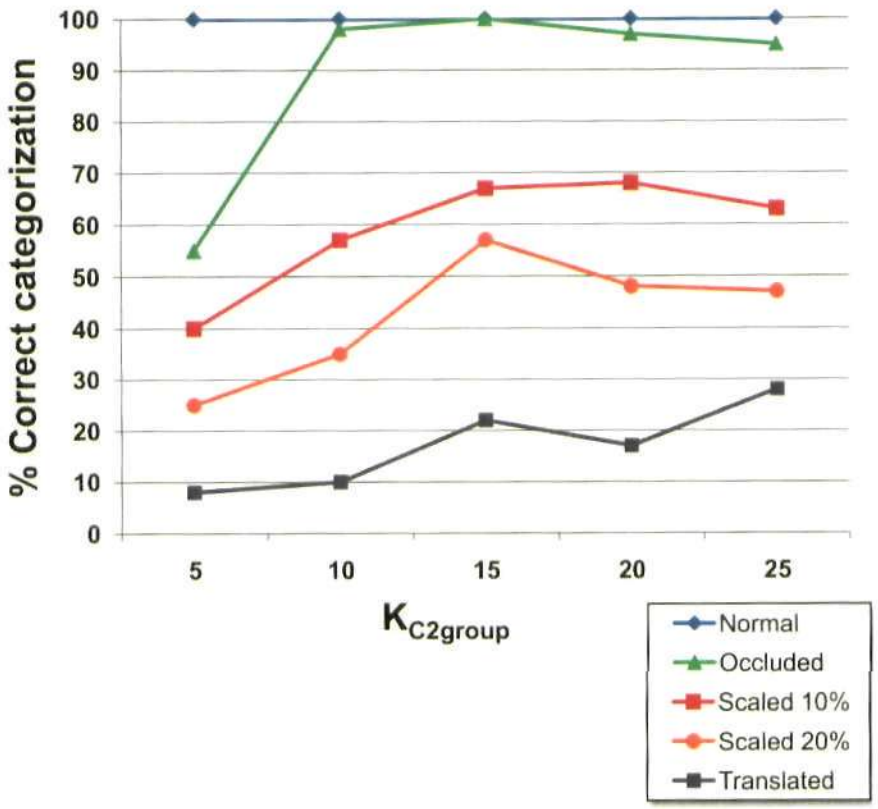


Figure 5.11: Categorization performance as a function of the number of states per group in the C2 layer, $K_{C2group}$, for the 3-level architecture.

5.1.2.2 Categorization as a function of the number of non-zero elements in the S2-C2 weight matrix

Figures 5.12, 5.13, 5.14 and 5.15 show the categorization performance as a function of the number of non-zero elements in the S2-C2 weight matrix (see Section 4.3 for details), for the four different S2 RF sizes. Results were obtained for the alternative 3-level architecture using values of $K_{C1group} = K_{C1group} = 10$, and are plotted for the five different test datasets as detailed in the figure legend.

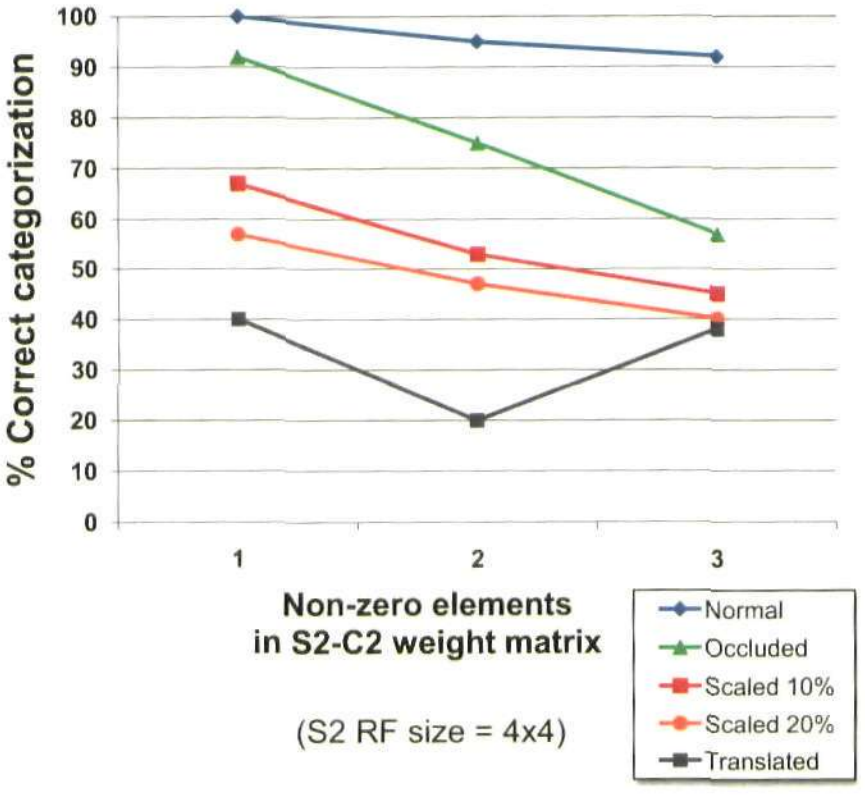


Figure 5.12: Categorization performance as a function of the number of non-zero elements in the S2-C2 weight matrix for the alternative 3-level architecture, using S2 RF size = 4x4.

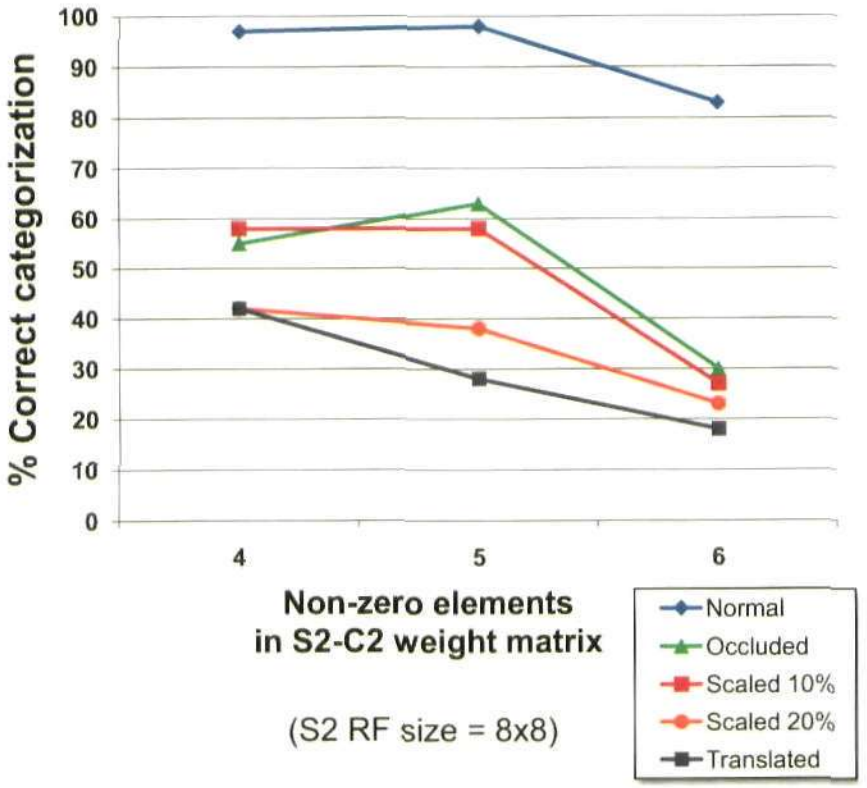


Figure 5.13: Categorization performance as a function of the number of non-zero elements in the S2-C2 weight matrix for the alternative 3-level architecture, using S2 RF size = 8x8.

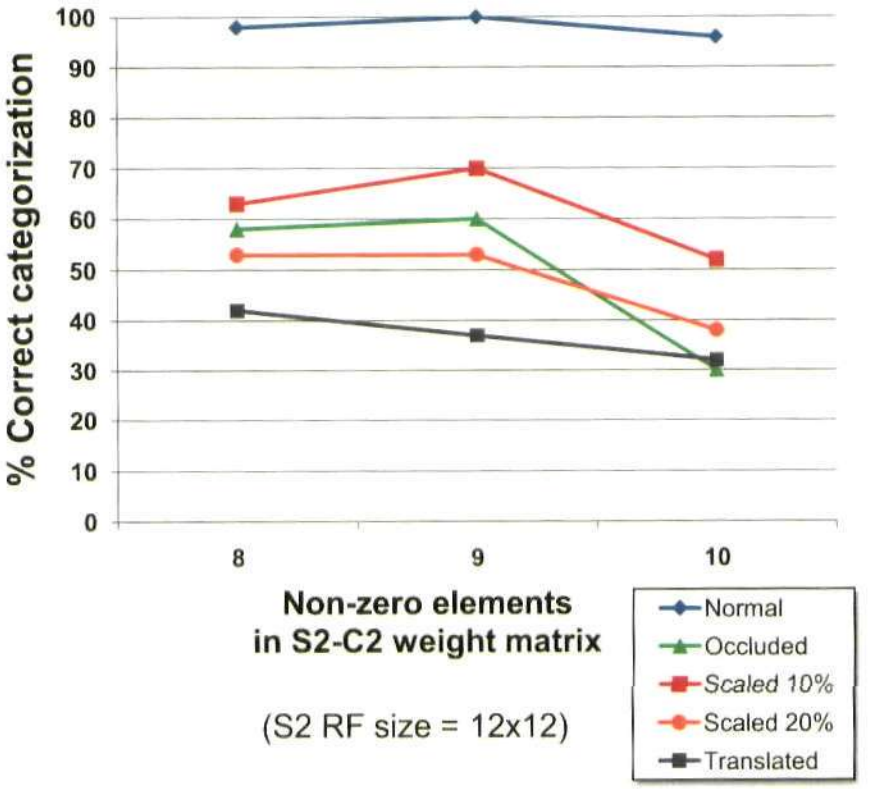


Figure 5.14: Categorization performance as a function of the number of non-zero elements in the S2-C2 weight matrix for the alternative 3-level architecture, using S2 RF size = 12x12.

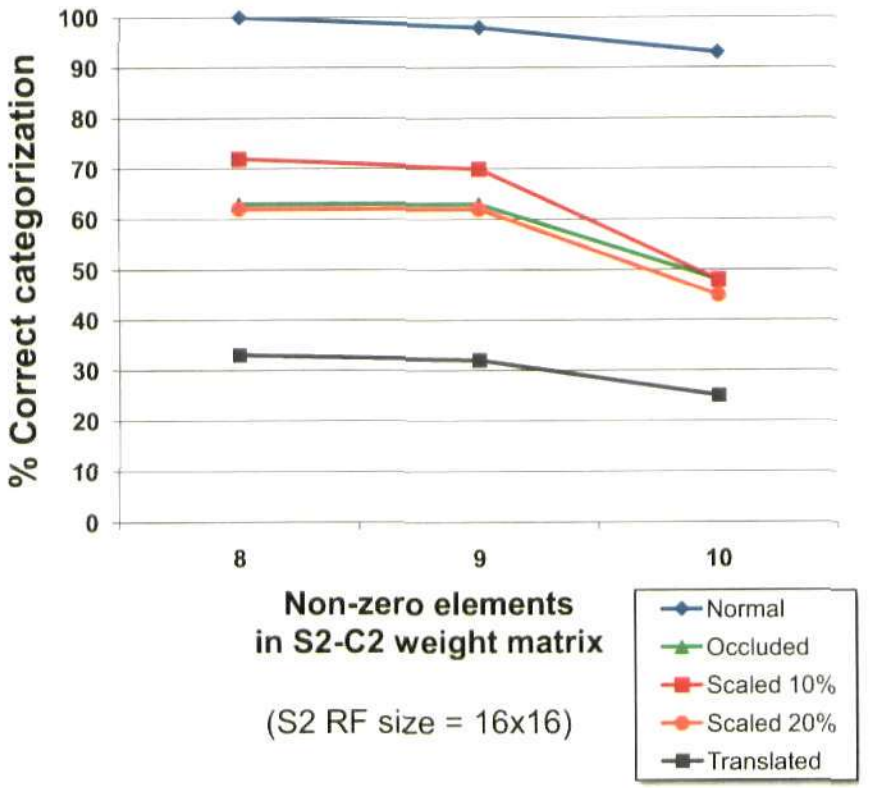


Figure 5.15: Categorization performance as a function of the number of non-zero elements in the S2-C2 weight matrix for the alternative 3-level architecture, using S2 RF size = 16x16.

5.1.2.3 Categorization as a function of the S2 RF size

Figure 5.16 shows the categorization performance as a function of the S2 RF size, which takes the values 4x4, 8x8, 12x12 and 16x16. The results shown were obtained using the alternative 3-level architecture and using values of $K_{C1group} = K_{C1group} = 10$ and the number of non-zero elements that maximized the performance for each S2 RF size: 1, 4, 8 and 8 respectively. Results are plotted for the five different test datasets as detailed in the figure legend.

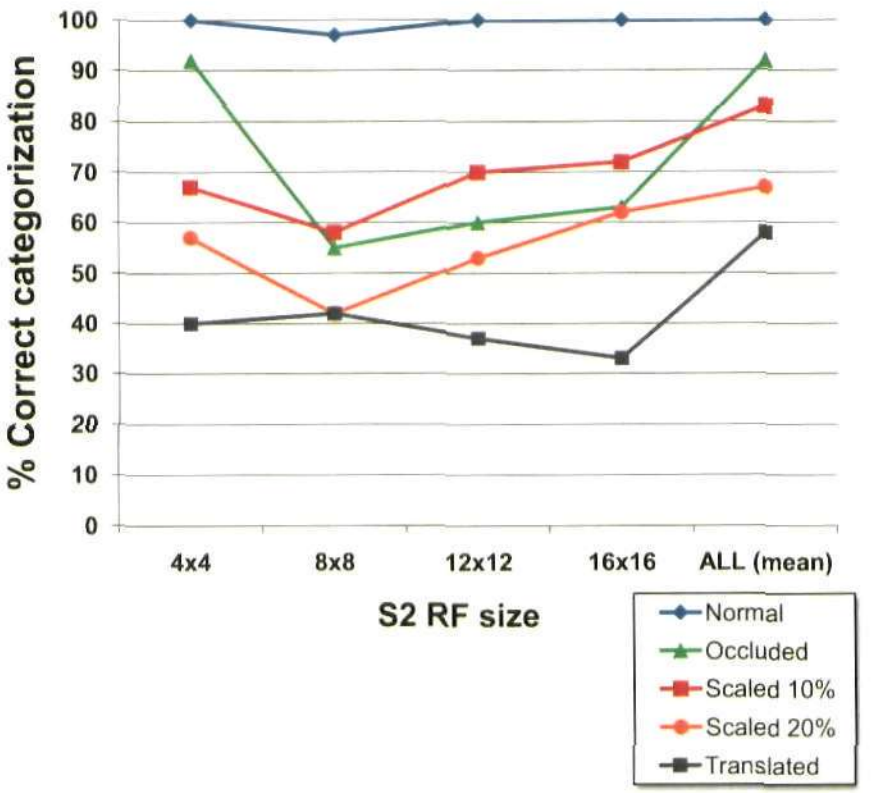


Figure 5.16: Categorization performance as a function of the S2 RF size for the alternative 3-level architecture. The rest of parameters were set as follows: $K_{C1group} = K_{C1group} = 10$ and the number of non-zero elements = 1, 4, 8, 8, respectively.

5.1.2.4 Comparison of different models

Figure 5.17 compares the categorization performance of the three versions of the model proposed, namely the 3-level architecture, the 4-level architecture and the alternative 3-level architecture, the HMAX model and an HTM network. For the 4-level architecture only the results for the normal dataset were calculated, as its poor performance suggested the results on the transformed datasets would be extremely low and thus not worth the computational cost.

The HMAX-like model was implemented using Matlab and replicates the model described in Serre et al. (2007c), i.e. the 3-level HMAX implementation. Following the original HMAX implementation, the S2 prototypes are selected at random from the training set, as opposed to employing the *minimum-distance* algorithm implemented in the Bayseian network model (see Section 4.3). The HTM-like results were obtained using the Numenta Vision Toolkit (George and Hawkins 2009), which allows one to train and test an HTM network. However, only 50 categories are allowed, so 10 categories had to be eliminated from the training and testing datasets.

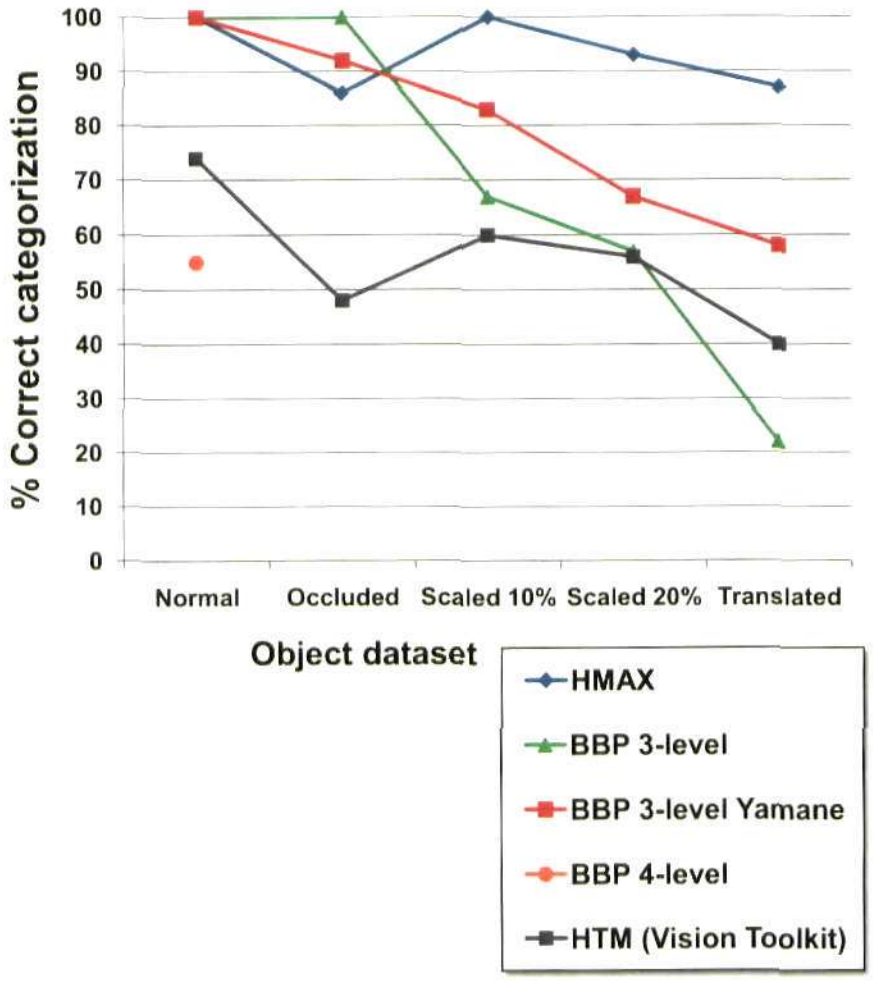


Figure 5.17: Comparison of categorization performance by the three versions of the model proposed, namely the 3-level architecture, the 4-level architecture and the alternative 3-level architecture, the HMAX model and an HTM network. For the 4-level architecture only the results for the normal dataset were obtained. The HMAX-like model was implemented using Matlab and replicates the model described in Serre et al. (2007c), i.e. the 3-level HMAX implementation. The HTM-like results were obtained using the Numenta Vision Toolkit (George and Hawkins 2009) which allows one to train and test an HTM network. Note for this graph the different object datasets are plotted along the x-axis, while the different models are shown with independent line graphs as detailed in the figure legend.

5.2 Feedback-mediated illusory contour completion

This section describes the model responses after feedback modulation has taken effect. This illustrates the interaction between the feedforward and feedback information in the network, reflected in the evolution over time of the belief at the different layers. The scenario chosen to illustrate these results consists of using the Kanizsa square as the input image and feeding back the representation of a square from higher layers. This section is structured to show the effects of feedback arising from progressively higher layers. In other words, the first set of results illustrates the simplest case, when feedback originates in the C1 layer, while the final set shows results for feedback originating in the top layer, S3, and targeting all inferior layers, including S1. All the results in this section were obtained using the alternative 3-level architecture, as this provided the best feedforward recognition results.

5.2.1 Feedback from C1 to S1

Figure 5.18 shows the S1 model response to a Kanizsa square input image while the C1 layer is *clamped* to a square representation. Thus, the results illustrate how the bottom-up evidence from the input image, $\lambda(S1)$, is combined with top-down information from the C1 layer, $\pi(S1)$.

5.2.2 Feedback from S2 to S1

Figure 5.19 shows the S1 and C1 model responses to a Kanizsa square input image while the S2 layer is *clamped* to a square representation. Thus, the results illustrate how the bottom-up evidence from the input image, $\lambda(S1)$, is combined with top-down information from the S2 layer, $Bel(S2)$, and how the representation at the S1 and C1 layers evolves over time.

Figure 5.20 shows the temporal response of the S1 and C1 belief for the region corresponding to the top horizontal illusory contour of the Kanizsa figure for the setup depicted in Figure 5.19.

Figure 5.21 compares the C1 belief responses as a function of the S2 RF size and the scale band for the setup shown in Figure 5.19.

Figure 5.22 compares the S1 and C1 model responses to the occluded Kanizsa and blurred Kanizsa input images at times t=1 and t=4, for the setup depicted in Figure 5.19.

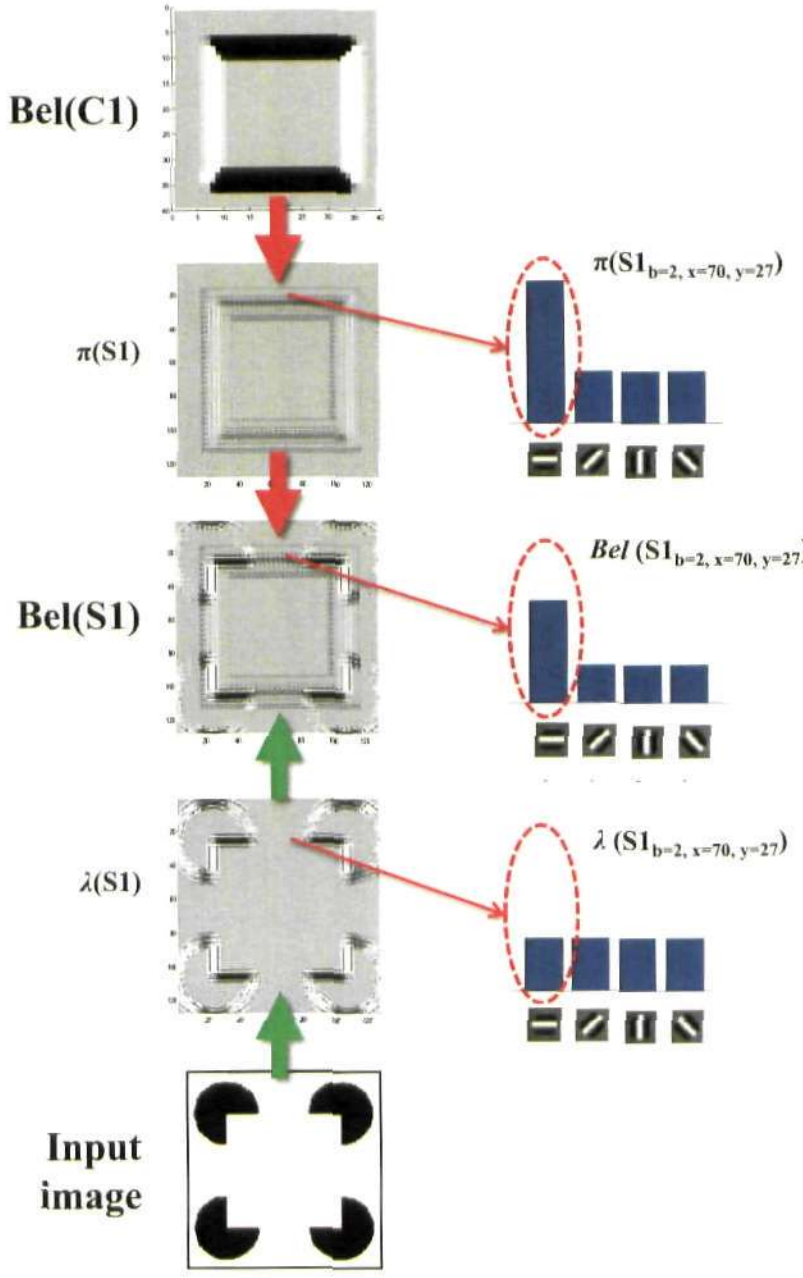


Figure 5.18: S1 model response to a Kanizsa square input image with feedback arising from the C1 layer containing a square representation. The 2D S1 maps represent the probability value for the horizontal state across all locations of scale band 2. Arrows indicate feedforward (green) and feedback (red) propagation of evidence. The probability distributions of an S1 node in the illusory contour region are shown on the left, illustrating how the bottom-up evidence, $\lambda(S1)$, and top-down information from the C1 layer, $\pi(S1)$, are combined to form the belief, $Bel(S1)$.

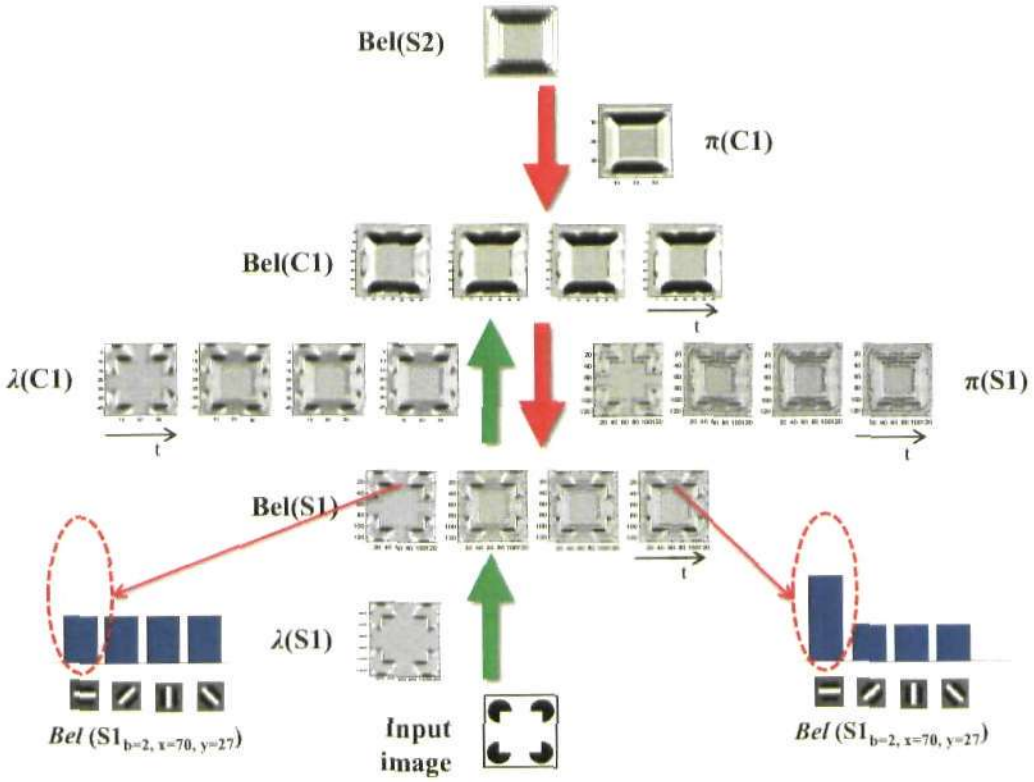


Figure 5.19: S1 and C1 model responses to a Kanizsa square input image with feedback arising from the S2 layer containing a square representation. The 2D S1 and C1 maps represent the probability value for the horizontal state across all locations of scale band 2 and 1, respectively. The S2 representation corresponds to the C1 reconstruction using the C1 features of the maximum S2 prototypes at each location, as described in Section 5.1. For each layer, the temporal evolution of the response from time t=1 to t=4 is shown, except for $\lambda(S1)$ and $\pi(C1)$ whose response is fixed over time. Arrows indicate feedforward (green) and feedback (red) propagation of evidence. The probability distributions of an S1 node in the illusory contour region at t=1 and t=4 are shown at the bottom of the figure, illustrating how the recurrent interaction between feedforward and feedback leads to an increase of the horizontal orientation belief.

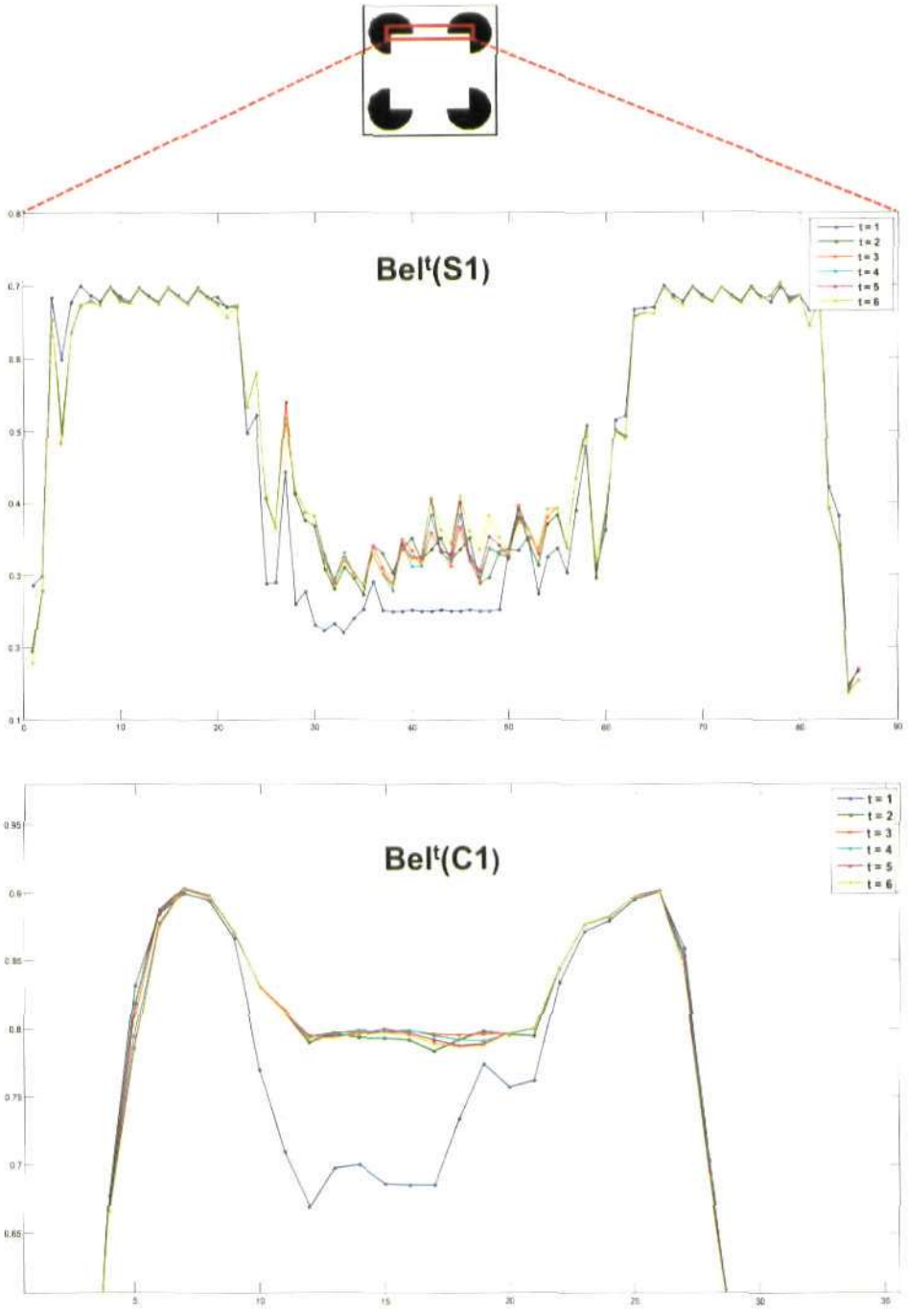


Figure 5.20: Temporal response of the S1 and C1 belief for the region corresponding to the top horizontal illusory contour of the Kanizsa figure. Feedback originates from the square representation in layer S2, as illustrated in the setup of Figure 5.19. More precisely, the response corresponds to S1 nodes at locations $\{x,y\} = \{25:28, 24:109\}$ and C1 nodes at locations $\{x,y\} = \{7:10, 6:36\}$ both averaged over the vertical dimension. The responses at times t=1 to t=6 are plotted in different colours as illustrated in the legend.

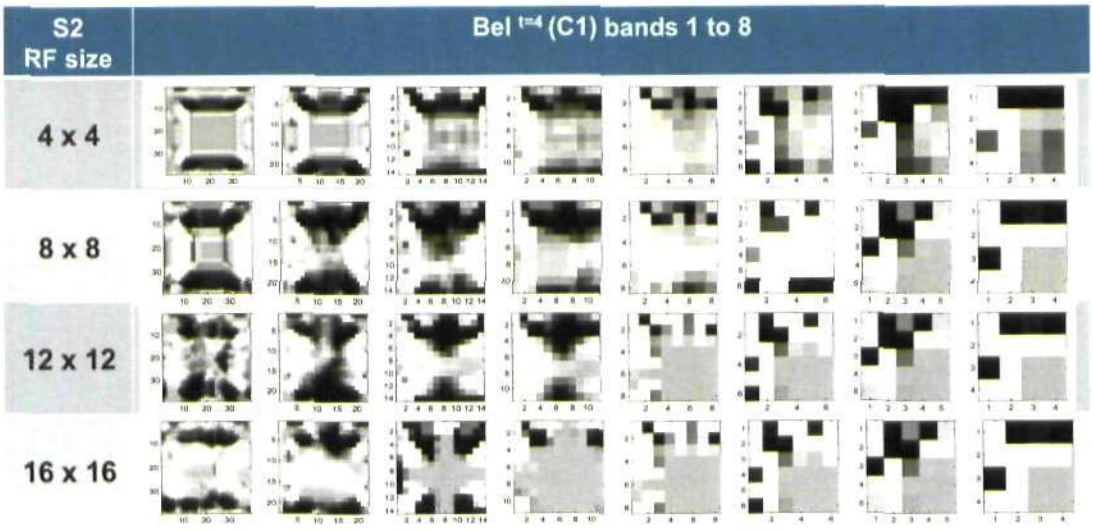


Figure 5.21: Comparison of the C1 belief responses at t=4, as a function of the S2 RF size and the scale band, to a Kanizsa square input and feedback arising from a square representation in layer S2. The experimental setup for these results is shown in Figure 5.19.

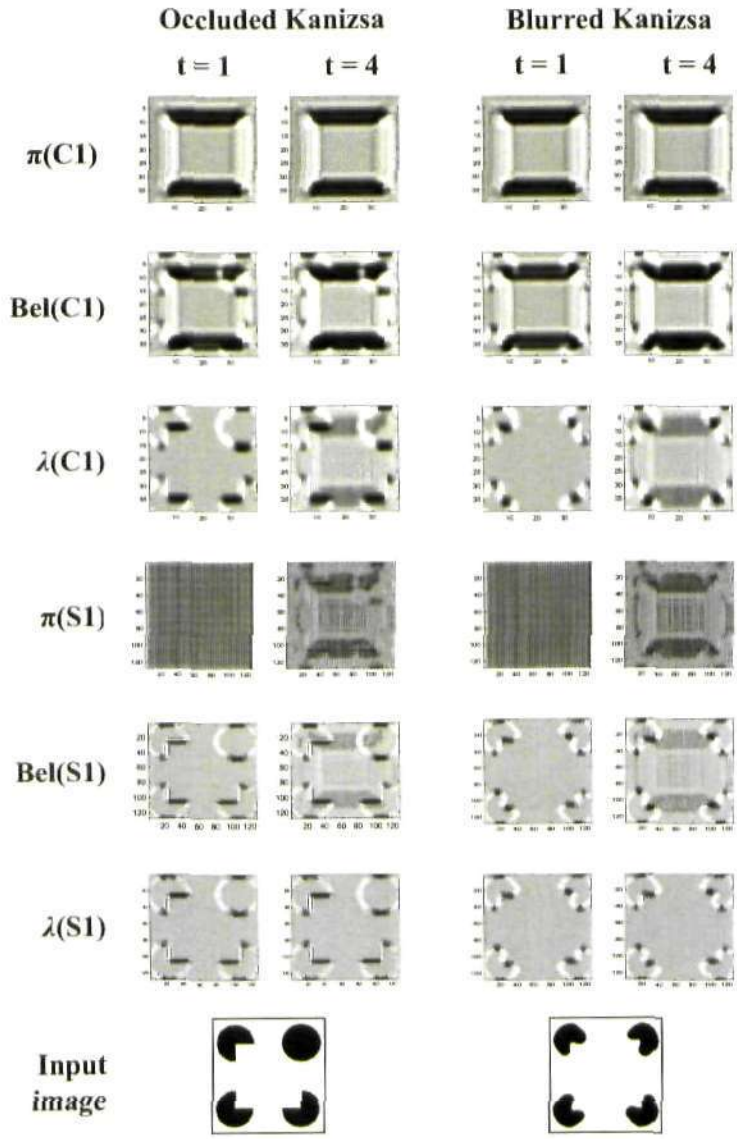


Figure 5.22: Comparison of the S1 and C1 model responses to the occluded Kanizsa and blurred Kanizsa input images at times t=1 and t=4. The experimental setup for this experiment is depicted in Figure 5.19 and corresponds to feedback originating from the square representation in layer S2.

5.2.3 Feedback from C2 to S2

Figure 5.23 compares the feedback generated by a square representation in the C2 layer as a function of the number of non-zero elements in the S2-C2 weight matrix and the sampling parameters N_{C2} and K_{C2}. In order to objectively compare the quality of the feedback reconstruction, we calculate the mean absolute difference between the C1 reconstruction, $\pi(C1)$, using the different S2-C2 weight matrices, and the ideal C1 square representation.

Figure 5.23 tests the influence of two factors on the model's capacity to perform feedback reconstruction: 1) the number of non-zero elements in the S2-C2 weight matrix during feedback, and 2) the sampling parameters. In order to test the influence of a third factor, namely the number of non-zero elements in the S2-C2 weight matrix used to generate the initial C2 square representation, two different sets of results are shown. The C2 square representation in Figure 5.23 was obtained using an S2-C2 weight matrix with one non-zero element, whereas the C2 square representation in Figure 5.24 was obtained using an S2-C2 weight matrix with two non-zero elements.

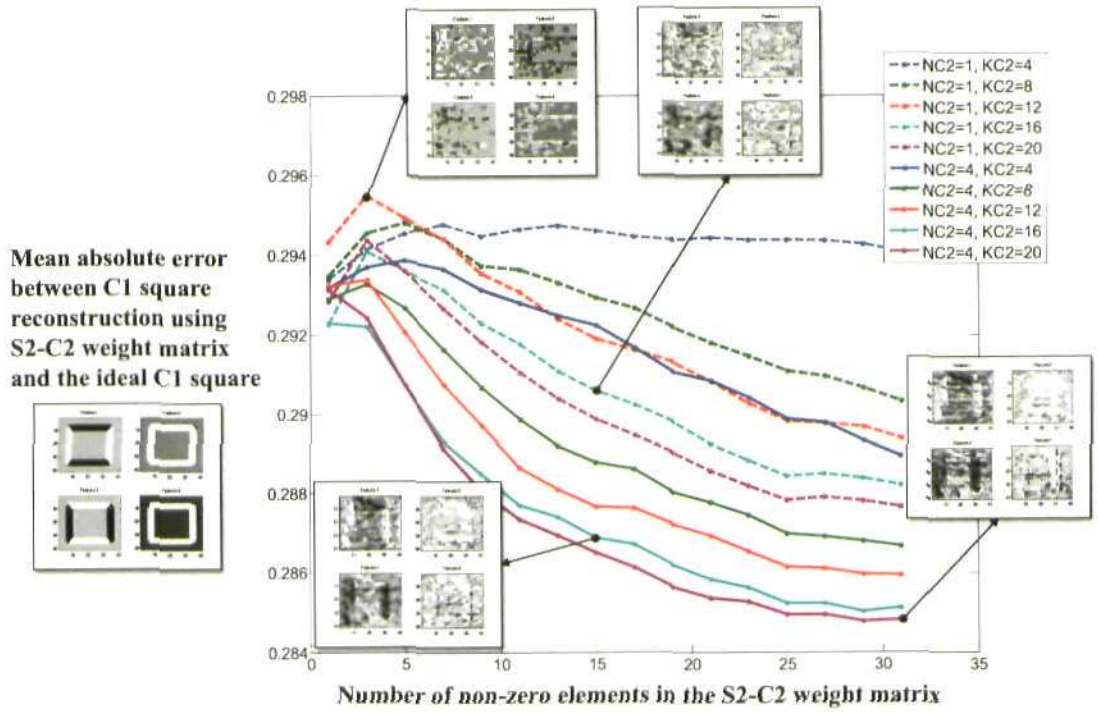


Figure 5.23: Comparison between the feedback generated by a square representation in the C2 layer as a function of the number of non-zero elements (x-axis) in the S2-C2 weight matrix and the sampling parameters N_{C2} and K_{C2} (different line graphs as shown in figure legend). The C2 representation was obtained using an S2-C2 weight matrix with one non-zero element. The y-axis corresponds to the mean square difference between the C1 reconstruction using the different S2-C2 weight matrices and the ideal C1 square representation for all nodes of C1, scale band 1. The C1 reconstruction, $\pi(C1)$, is obtained exclusively from the $\pi(S2)$ response, such that no feedforward likelihood function is involved, using the fixed sampling parameters $N_{C1} = 4$ and $K_{C1} = 4$. The ideal C1 square representation is shown underneath the y-axis label. Three of the C1 reconstructions from S2-C2 weight matrices with different parameters are also shown to visually illustrate that lower error values generally correspond to C1 reconstructions closer to the ideal C1 square.

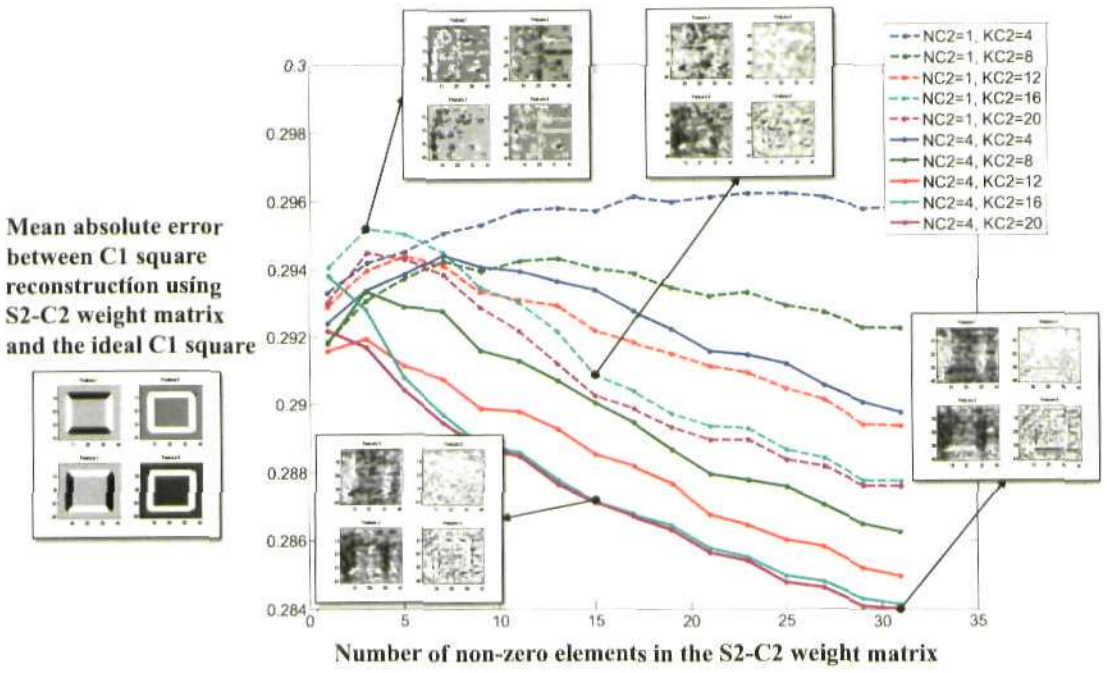


Figure 5.24: Comparison between the feedback generated by a square representation in the C2 layer as a function of the number of non-zero elements (x-axis) in the S2-C2 weight matrix and the sampling parameters N_{C2} and K_{C2} (different line graphs as shown in figure legend). The C2 representation was obtained using an S2-C2 weight matrix with two non-zero element. The y-axis corresponds to the mean square difference between the C1 reconstruction using the different S2-C2 weight matrices and the ideal C1 square representation for all nodes of C1, scale band 1. The C1 reconstruction, $\pi(C1)$, is obtained exclusively from the $\pi(S2)$ response, such that no feedforward likelihood function is involved, using the fixed sampling parameters $N_{C1} = 4$ and $K_{C1} = 4$. The ideal C1 square representation is shown underneath the y-axis label. Three of the C1 reconstructions from S2-C2 weight matrices with different parameters are also shown to visually illustrate that lower error values generally correspond to C1 reconstructions closer to the ideal C1 square.

5.2.4 Feedback from C2 to S1

Figure 5.25 shows the S1, C1 and S2 model responses to a Kanizsa square input image while the top-down feedback from C2, $\pi(S2)$, is *clamped* to a square representation. Thus, the results illustrate how the bottom-up evidence from the input image, $\lambda(S1)$ are combined with top-down information from the C2 layer, $\pi(S2)$, and how the representation at the S1, C1 and S2 layers evolves over time. Note that in this case feedback does not arise directly from the belief in the C2 layer, $Bel(C2)$, because as illustrated in the previous subsection, it is difficult to obtain a good $\pi(S2)$ square reconstruction via the S2-C2 weight matrix. Instead, the $\pi(S2)$ is fixed to an ideal S2 square representation in order to test the capacity of the model to combine feedback information hypothetically generated from the C2 layer.

Figure 5.26 shows the temporal response of the S1 and C1 belief for the region corresponding to the top horizontal illusory contour of the Kanizsa figure. The responses shown are for the setup depicted in Figure 5.25, where the square representation is fed back from $\pi(S2)$.

Figures 5.27 5.28 compares the S1 and C1 belief responses to the occluded Kanizsa, blurred Kanizsa and blank input images at times t=2 and t=8, for the setup depicted in Figure 5.19 where the square representation is fed back from $\pi(S2)$.

Figure 5.29 compares S1, C1 and S2 model responses, using the setup of Figure 5.25, for the three different belief update methods illustrated in Figure 4.19.

Figure 5.30 shows the S1, C1 and S2 model responses to a Kanizsa square input image while the C2 layer is *clamped* to a square representation. Thus, the results illustrate how the bottom-up evidence from the input image, $\lambda(S1)$ are combined with information from the belief in the C2 layer, $Bel(C2)$, and how the representation at the S1, C1 and S2 layers evolves over time. In this case feedback arises directly from the belief in the C2 layer, $Bel(C2)$, using the S2-C2 weight matrix with X non-zero elements and sampling parameters $N_{C2} = X$ and $K_{C2} = X$. These parameters were chosen to maximize the similarity between the square C1 feedback reconstruction from C2 and the ideal C1 square representation as depicted in Figure 5.23.

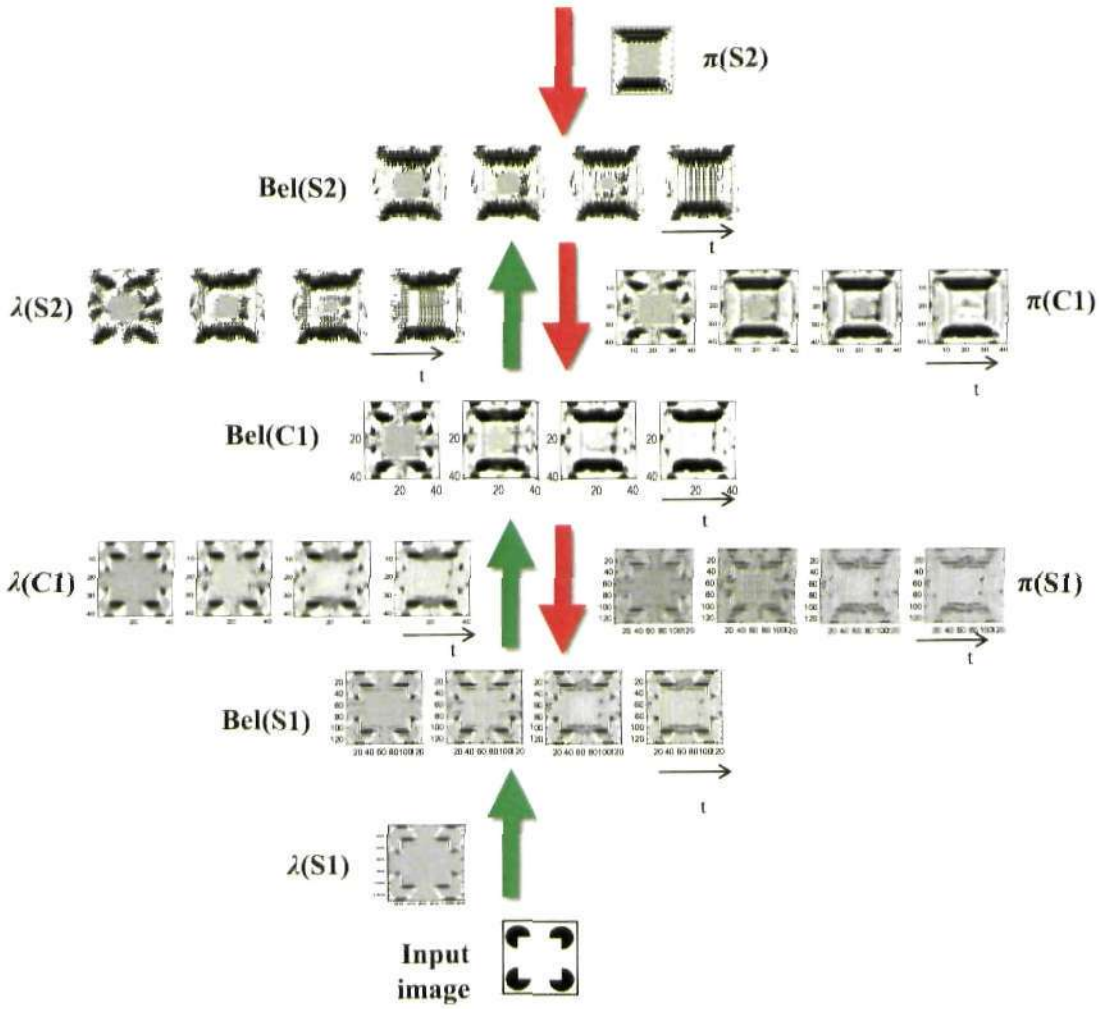


Figure 5.25: S1, C1 and S2 model responses to a Kanizsa square input image while the top-down feedback from C2, $\pi(S2)$, is *clamped* to a square representation. The 2D S1 and C1 maps represent the probability value for the horizontal state across all locations of scale band 2 and 1, respectively. The S2 representation corresponds to the C1 reconstruction using the C1 features of the maximum S2 prototypes at each location, as described in Section 5.1. For each layer, the temporal evolution of the response from time t=1 to t=4 is shown, except for $\lambda(S1)$ and $\pi(S2)$, whose responses are fixed over time. Arrows indicate feedforward (green) and feedback (red) propagation of evidence. Note that in this case feedback does not arise directly from the belief in the C2 layer, $Bel(C2)$, because as illustrated in the previous subsection, it is difficult to obtain a good $\pi(S2)$ square reconstruction via the S2-C2 weight matrix. Instead, the $\pi(S2)$ is fixed to an ideal S2 square representation in order to test the capacity of the model to combine feedback information hypothetically generated from the C2 layer.

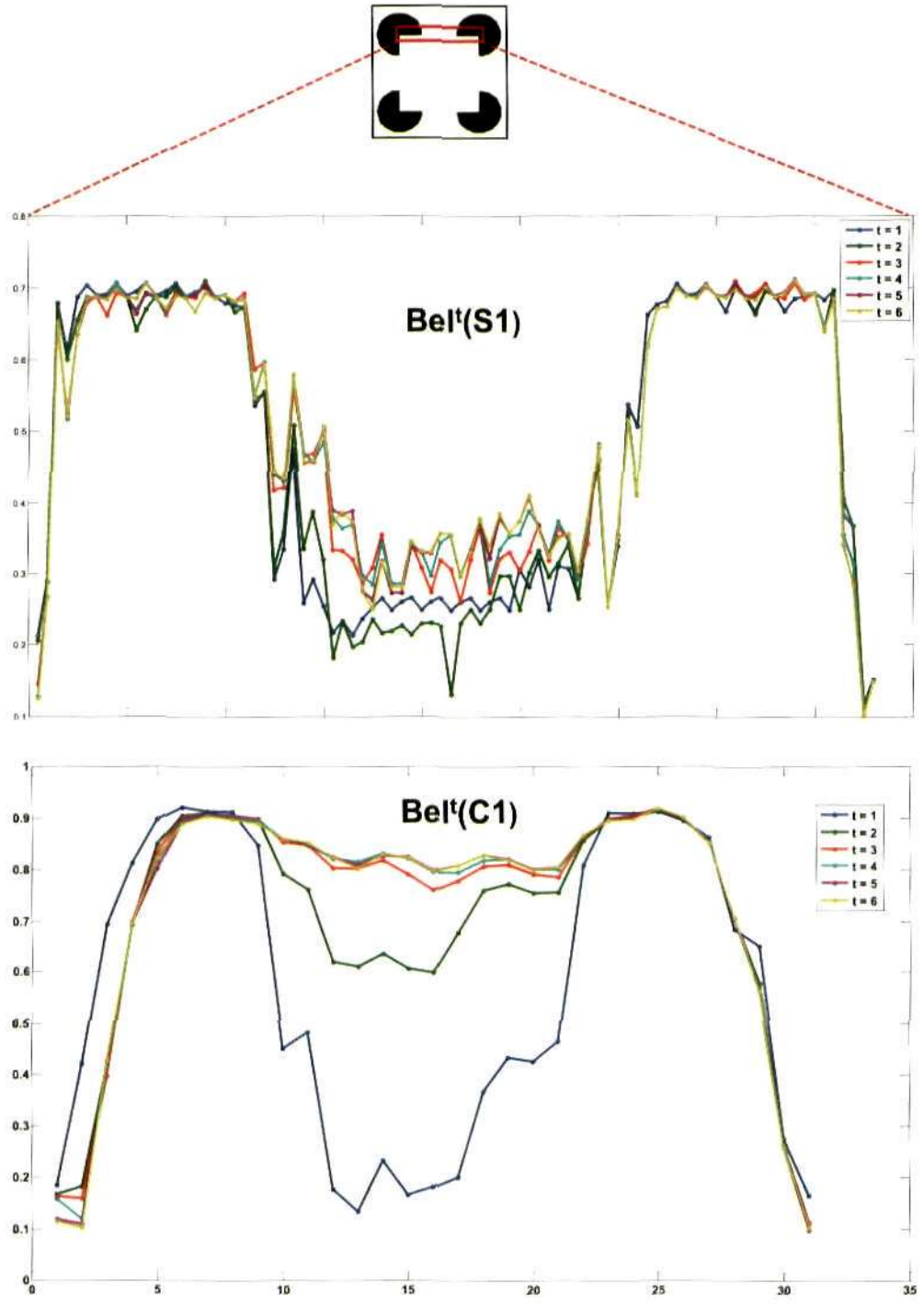


Figure 5.26: Temporal response of the S1 and C1 belief for the region corresponding to the top horizontal illusory contour of the Kanizsa figure. Feedback originates from the square representation fed back via the $\pi(S2)$, as illustrated in Figure 5.25. More precisely, the response corresponds to S1 nodes at locations $\{x,y\} = \{25:28, 24:109\}$ and C1 nodes at locations $\{x,y\} = \{7:10, 6:36\}$, both averaged over the vertical dimension. The responses at times t=1 to t=6 are plotted in different colours, as illustrated in the legend.

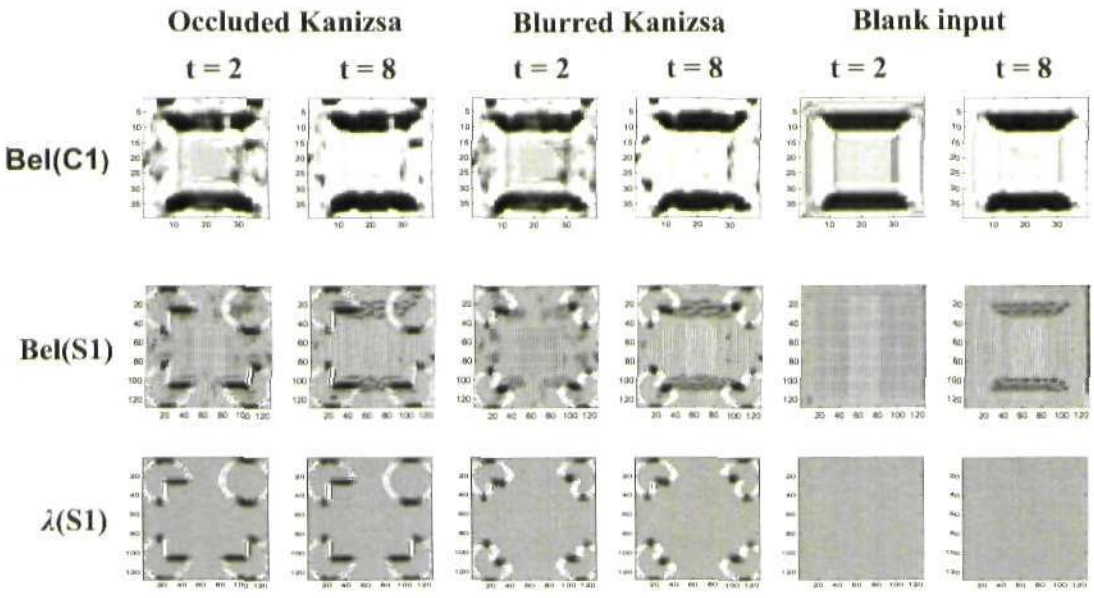


Figure 5.27: Comparison of the S1 and C1 model responses to the occluded Kanizsa, blurred Kanizsa and blank input images at times t=2 and t=8. The experimental setup for this experiment is depicted in Figure 5.25 and corresponds to feedback originating from the square representation in $\pi(S2)$.

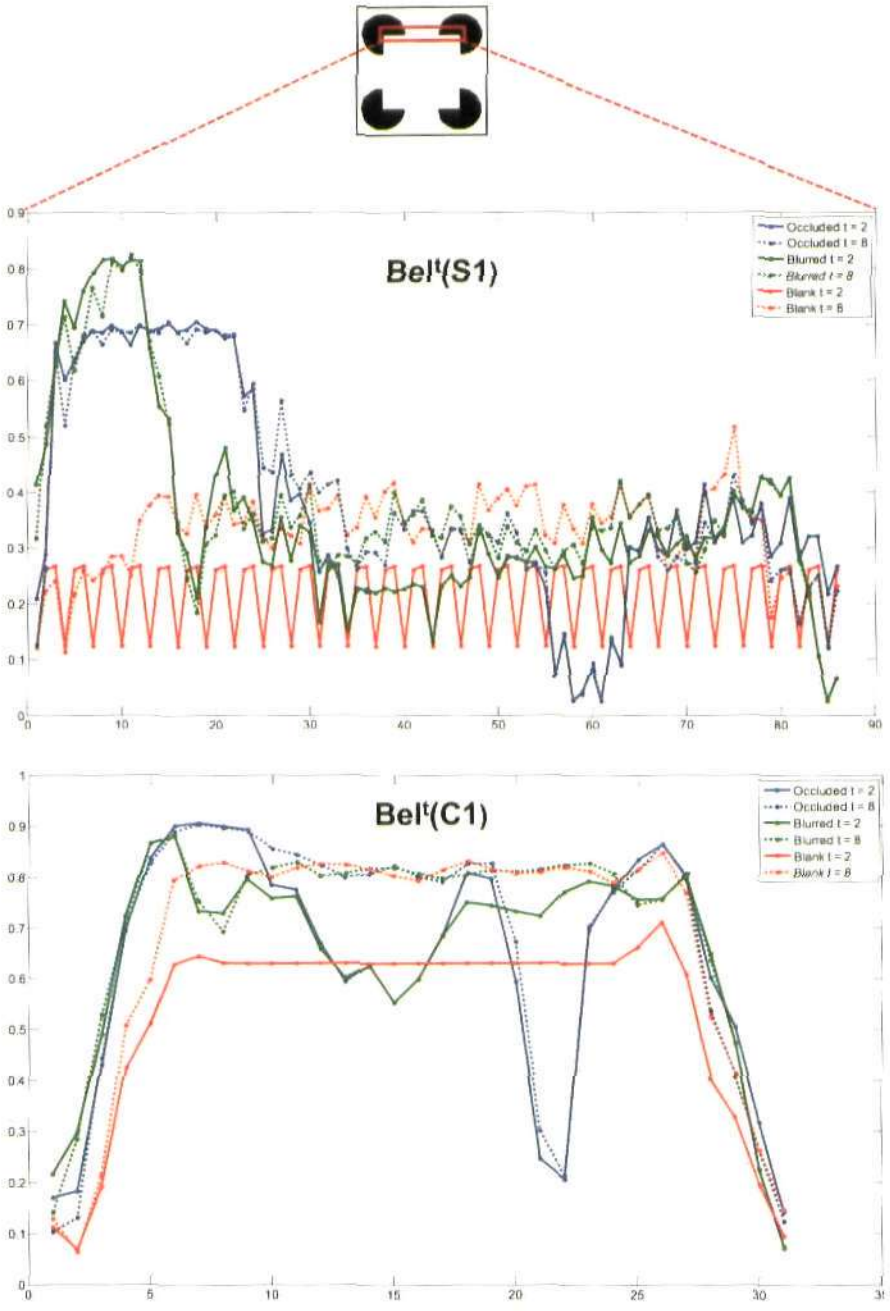


Figure 5.28: Temporal response of the S1 and C1 belief for the region corresponding to the top horizontal illusory contour of the Kanizsa figure for the occluded Kanizsa, blurred Kanizsa and blank input images. Feedback originates from the square representation fed back via the $\pi(S2)$ as illustrated in the setup of Figure 5.25. More precisely the response corresponds to S1 nodes at locations $\{x,y\} = \{25:28,24:109\}$ and C1 nodes at locations $\{x,y\} = \{7:10,6:36\}$, both averaged over the vertical dimension. The response at times t=2 (unbroken line) and t=8 (dotted line) are plotted in different colours for the different input images, as illustrated in the legend.

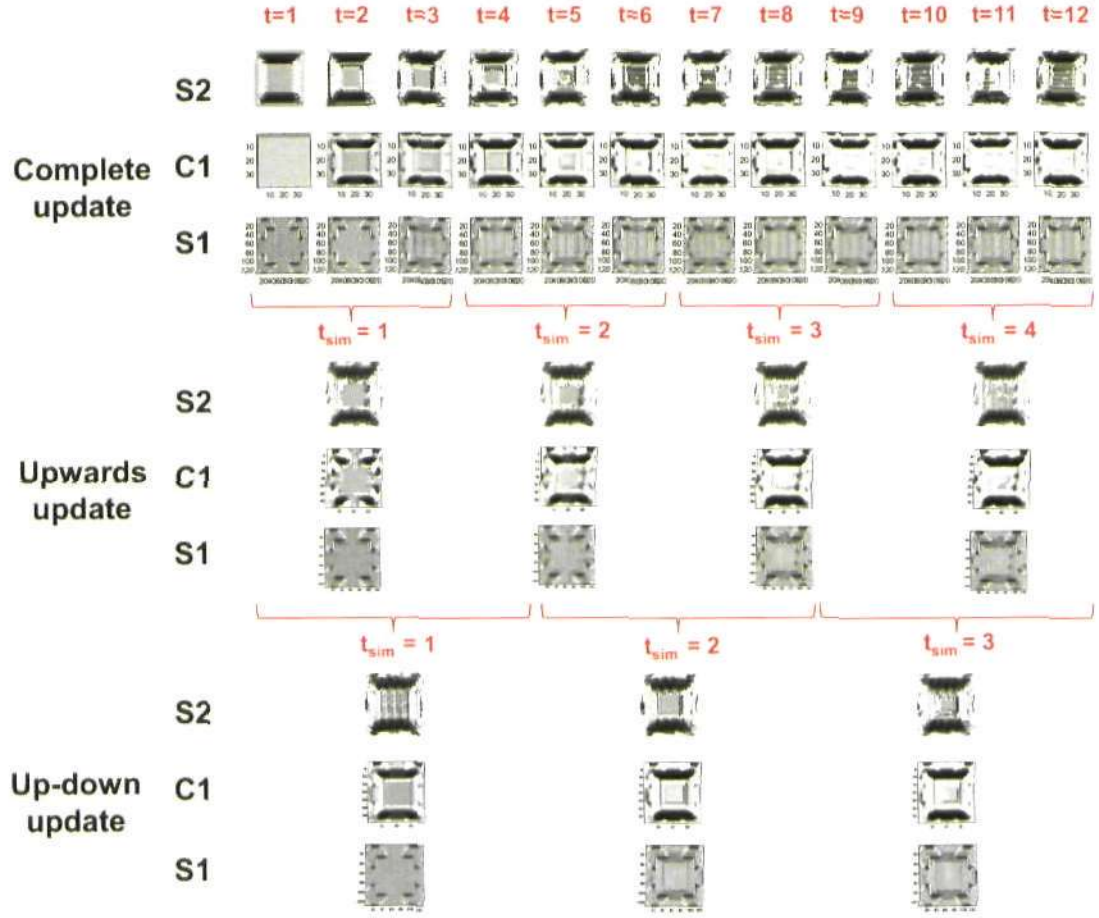


Figure 5.29: Comparison of S1, C1 and S2 model responses, using the setup of Figure 5.25, for the three different belief update methods illustrated in Figure 4.19. The *complete* method updates all layers at every time step. The *upwards* method updates one layer per time step in ascending order, thus each simulation time step, t_{sim}, is equivalent to three (the number of layers updated) original time steps. The *up-down* method updates one layer per time step in ascending order until it reaches the top layer and later in descending order, thus each simulation step, t_{sim}, is equivalent to five ($2 \times$ number of layers -1) original time steps.

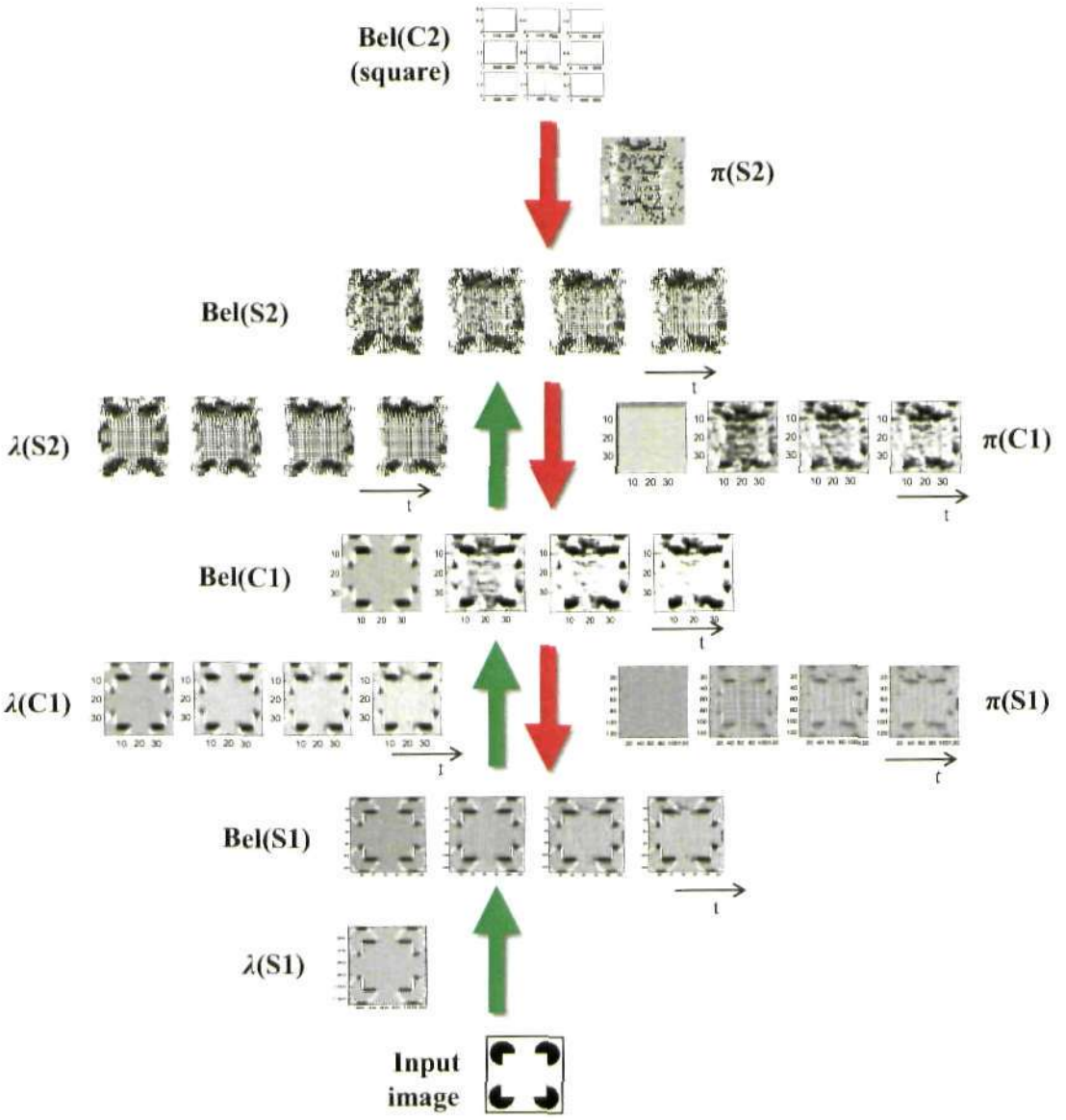


Figure 5.30: S1, C1 and S2 model responses to a Kanizsa square input image while the C2 layer is *clamped* to a square representation. These results illustrate how the bottom-up evidence from the input image, $\lambda(S1)$, is combined with information from the belief in C2 layer, $Bel(C2)$, and how the representation at the S1, C1 and S2 layers evolves over time. The square representation in the C2 layer, $Bel(C2)$, is fed back using the S2-C2 weight matrix with 23 non-zero elements and sampling parameters $N_{C2} = 4$ and $K_{C2} = 6$. These parameters were chosen to maximize the similarity between the square C1 feedback reconstruction from C2 and the ideal C1 square representation as depicted in Figure 5.23.

5.2.5 Feedback from S3 to S1

Figure 5.31 shows the S1, C1, S2 and C2 model responses to a Kanizsa square input image while the S3 layer is *clamped* to a square representation. Thus, the results illustrate how the bottom-up evidence from the input image, $\lambda(S1)$, is combined with information from the belief in the S3 layer, $Bel(S3)$, and how the representation at the S1, C1, S2 and C2 layers evolves over time. In this case the *up-down* belief update method was implemented as it provided a cleaner response than the *upwards* method. This means the response is shown for two complete up and down cycles, starting and finishing at the S1 layer. For this reason the response for the S1 layer only shows three time steps, corresponding to the up pass, wheres layers C1, S2 and C2 show four time steps, corresponding to the up and down passes. The results correspond to two complete simulation time steps, $t_{sim} = 1, 2$, plus the S1 layer response for the third simulation time step, $t_sim = 3$, as illustrated at the bottom of Figure4.19.

5.2.6 Object categorization with feedback

Figure 5.32 compares the ranking of the square prototype over the S3 layer belief distribution for different input images and model parameters. Results are shown for the four different S2 RF sizes as well as for the mean response. These results were obtained using the alternative 3-level architecture. Note that these results represent the categorization response after the initial time step or bottom-up pass, assuming flat initial distributions in all layers.

[1]Caption for Figure 5.31. S1, C1, S2 and C2 model responses to a Kanizsa square input image while the S3 layer is *clamped to a square representation. These results illustrate how the bottom-up evidence from the input image,* $\lambda(S1)$, is combined with information from the belief in S3 layer, $Bel(S3)$, and how the representation at the S1, C1, S2 and C2 layers evolves over time. The square representation in the S3 layer, $Bel(S3)$, is fed back using the S2-C2 weight matrix with 23 non-zero elements and sampling parameters $N_{C2} = 4X$ and $K_{C2} = 6$. These parameters were chosen to maximize the similarity between the square C1 feedback reconstruction from C2 and the ideal C1 square representation as depicted in Figure 5.23. Also, the *up-down* belief update method was employed in order to reduce the noise of the lower level responses. This means the response is shown for two complete up and down cycles, starting and finishing at the S1 layer.

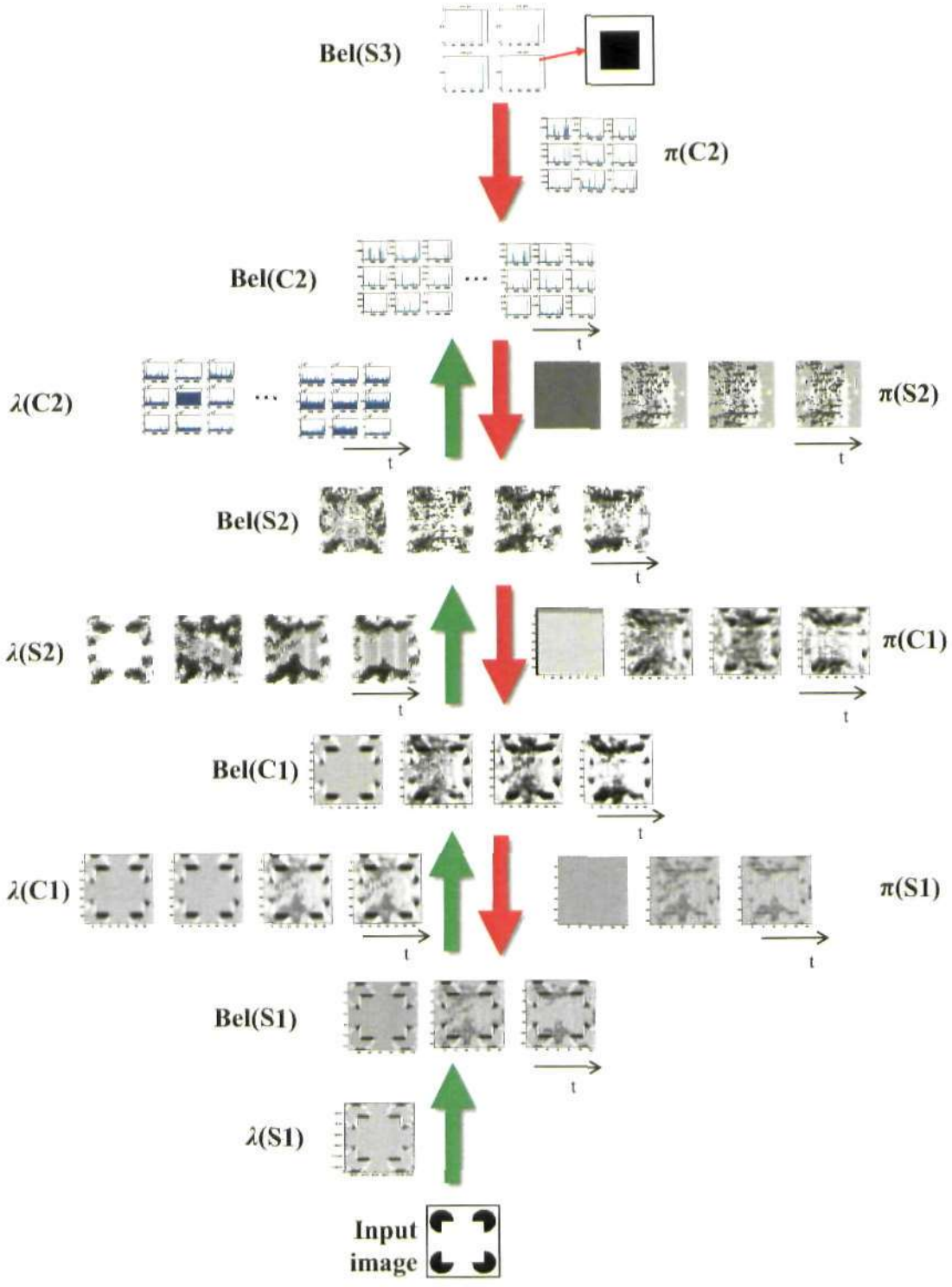


Figure 5.31: For caption see footnote[1].

Input image and setup	S2 RF size				
	4x4	8x8	12x12	16x16	ALL (mean)
Real square All bands No feedback	1	1	1	1	1
Real square All bands *Upwards* update (loopy)	1	9	2	1	1
Real square 1 band *Upwards* update (loopy)	1	>10	2	1	1
Real square 1 band *Complete* update (loopy)	4	-	-	-	-
Kanizsa square All bands No feedback	>10	>10	1	>10	1
Kanizsa square All bands *Upwards* update (loopy)	1	3	8	7	3
Kanizsa square 1 band *Upwards* update (loopy)	2	7	>10	6	4
Kanizsa square 1 band *Complete* update (loopy)	4	-	-	-	-

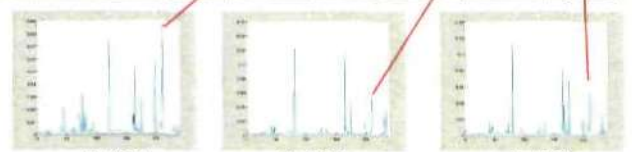


Figure 5.32: Ranking position of the square prototype over the S3 layer belief distribution for different input images and model parameters. Results are shown for the four different S2 RF sizes, as well as for the mean response. The input image is either a real square or a Kanizsa square. Each of them are compared for four conditions: 1) all bands and no feedback, 2) all bands and the *upwards* update method, 3) only the lower band and the *upwards* update method 4) only the lower band and the *complete* update method are used. The S3 belief distribution is shown for three of the results. Note, in the alternative Yamane 3-level architecture there are four prototypes or states per object, and all of these are considered a correct match when calculating the rank order. Importantly, the rank order is obtained by considering the states of each S3 node separately (there are 2 by 2 S3 nodes, one for each coded location), such that states from different S3 nodes compete with each other.

5.3 Feedback to S3: attention and priming

Figure 5.33 shows the model S3 belief response to an input image where a lamp occludes a dog, given two different S3 priors, $\pi(S3)$: an equiprobable or flat distribution and a biased distribution where the prior probability of objects that are animals has been doubled.

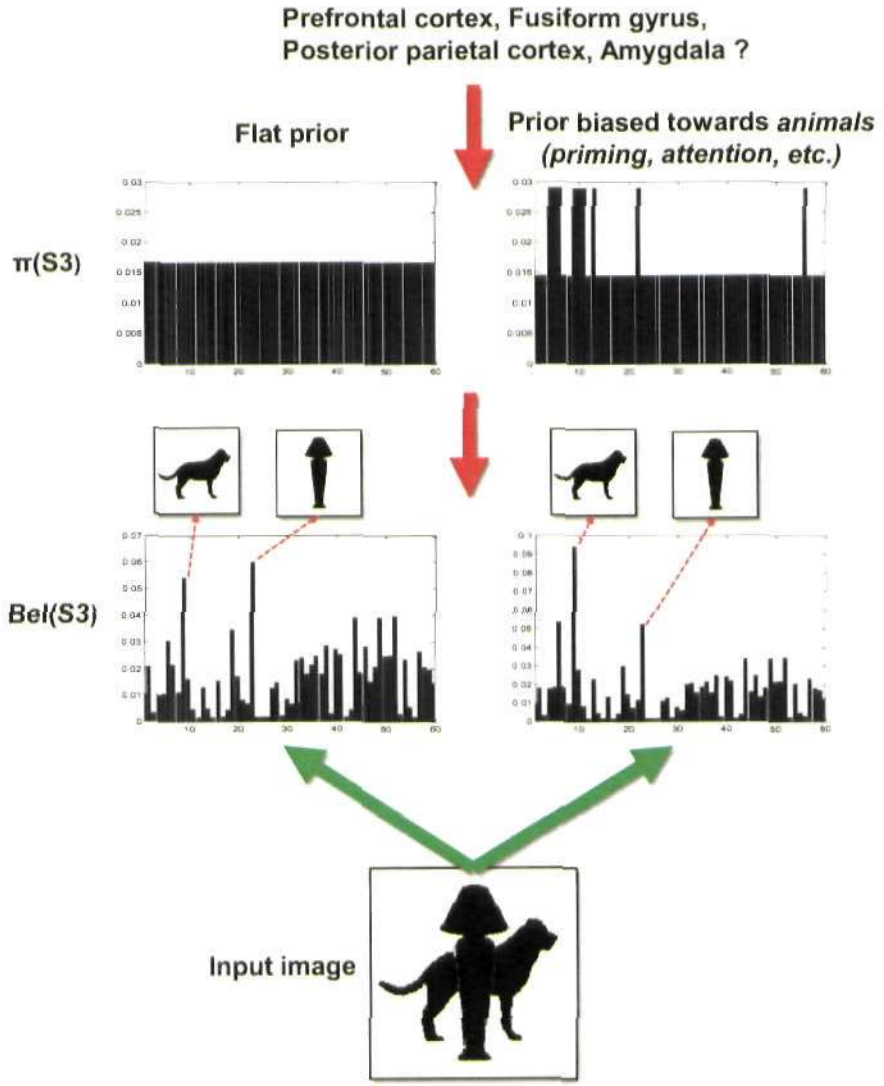


Figure 5.33: Comparison of S3 belief response to an input image containing a lamp occluding a dog, given two different S3 priors, $\pi(S3)$: an equiprobable or flat distribution and a biased distribution where the prior probability of objects that are animals has been doubled. The S3 prior is hypothesized to originate from regions outside of the ventral pathway. For the flat S3 prior the winner object in the S3 belief is the lamp, but for the biased prior where the *animal* objects are favoured, the winner object is the dog.

5.4 Original contributions in this chapter

- Simulation results illustrating the feedforward response of each of the layers in the proposed model.
- Simulation results of the model's object categorization performance as a function of different model parameters and in comparison with previous models.
- Simulation results showing the effects of feedback arising from the different layers of the model and how this achieves illusory contour completion. The performance is compared for several model parameters and belief update methods.
- Simulation results showing the effect of modifying the top layer prior on the categorization distribution.

Chapter 6

Discussion and conclusions

6.1 Analysis of results

6.1.1 Feedforward processing

6.1.1.1 Layer by layer response

The filtered image constitutes the input to the Bayesian network and is coded as the λ messages of a set of dummy nodes at all locations and scales, as shown in Figure 5.1. Each S1 node receives an input message from one of the dummy nodes and obtains a normalized probability distribution, $\lambda(S1)$, over the four states (orientations). Gabor filters have been widely used to model the response properties of V1 simple cells, including the preprocessing that occurs at the retina and lateral geniculate nucleus.

As illustrated in Figure 5.2, the S1 response is equivalent to that of the dummy nodes except that, due to normalization, blank input regions now present an equiprobable distribution such that each orientation has a value of 0.25. This can be understood as the background activity observed in non-active neural populations (Deneve 2008a). Furthermore, lateral inhibitory connections have been suggested to provide a normalization-like operation within pools of functionally similar neurons (Grossberg 2003, Kouh and Poggio 2008). Normalization has been associated with homeostatic functions crucial for stability and to maintain activity within an appropriate working regime (Grossberg 2003).

The C1 model response (Figure 5.3) shows a qualitatively similar pattern to the HMAX C1 response (Figure 5.4). The model response provides a lower resolution version of the input image, mimicking the *max* operation implemented in HMAX. The multiplicative combination of evi-

dence and a further normalization process in the model leads to a more radical representation of the orientations present at each location. While in HMAX the value of one orientation doesn't influence the values of the rest, the proposed model acts more like a winner-take-all network where a high valued state reduces the activity of the rest due to the sum-normalization operation. Further analysis is required to determine whether this approach might have advantages over the original HMAX by providing a potentially more discriminative and less redundant representation. Crucially, this approach doesn't prevent the C1 response from encoding the presence of two orientations at the same location, such as the vertical and diagonal orientations at the sides of the letter *A*, shown in Figure 5.3. It would be interesting to study the effects of lowering the the contrast of one of the orientations has on the C1 response. Given the normalization response across orientations, it is likely that the proportional response of the stronger to the weaker orientation is higher than that dictated by the relative contrast levels.

The S2 and C2 model representation (Figures 5.5 and 5.6) is more difficult to compare to that of the HMAX model because the number of prototypes is much higher (1000) and these have been learned from the distinct C1 responses corresponding to each model. However, the parameters in the models are equivalent and the selectivity and invariance operations are implemented in an analogous way to the S1 and C1 layers, which have been shown to provide good approximations. Furthermore, a good measure of whether the S2 and C2 layers reproduce the HMAX functionality is given by the categorization performance of the S3 layer, which relies on the C2 features.

The model S3 layer, shown in Figure 5.7, differs from the HMAX top layer in that it is integrated in the Bayesian network and implements the same selectivity operation as lower layers, whereas in HMAX the top layer is implemented as a separate support vector machine (SVM). The input to the SVM classifier consists of all the C2 features of the four different S2 RF sizes. In the proposed model, the features with different S2 RFs are processed along four parallel routes, each providing an S3 categorization response that is averaged to yield the overall response. This allows one to compare the categorization performance of the model for each of the individual S2 RF sizes, as well as for all of them combined.

Another particularity of the proposed model is the use of S3 nodes at four different locations, each containing four different prototypes for each object. Each prototype corresponds to the object positioned at one of the four potential locations as illustrated in Figure 5.7. It is straight forward to see why this approach leads to an improvement in the translated test-set categorization results. An image is correctly categorized if the average value over the four positions and four prototypes of the corresponding object is the highest. Although the averaging procedure is not explicitly included in the Bayesian network, it can be trivially implemented using the output of the S3 layer as the input to a simple linear classifier.

6.1.1.2 Object categorization

The set of objects shown in Figure 5.8 was chosen to have similar characteristics to the square shape, which was the key object required to demonstrate the Kanizsa figure contour completion effect. This was chosen over a standard object recognition test bench as the model focuses on the integration of information to achieve perception and not on improving the categorization performance of previous models. Despite not being a standard test bench, the same training and testing datasets (see Figure 5.9) were used when comparing the results to those of previous models such as HMAX and HTM. Future comparisons using standard test benches are possible, as the model can be tested with any arbitrary set of images simply by learning the weights from the corresponding training set.

It is important to note that although the objects present relatively simple forms, the fact that they are silhouettes increases the categorization difficulty as there is no gray-scale information within the object. Moreover, a previous model that was tested using the same 60 silhouette objects, as well as using gray-scale natural images, produced a similar categorization performance for both datasets (Dura-Bernal et al. 2010).

The first set of categorization results shows the effect of the C1 and C2 states per group. The concept of grouped states is introduced in order to approximate the invariance operation, as described in Sections 4.3 and 4.4. For each C1 group a set number of features (states) is learned from the image statistics. Each C1 state represents the product of the response of several S1 nodes to the same feature.

The graph in Figure 5.10 suggests that the optimum value for $K_{C1group}$ is approximately 10, while the graph in Figure 5.11 suggests that the optimum value for $K_{C2group}$ is approximately 15. If there are not enough C1 features per group, some input spatial arrangements of S1 nodes will not be captured, decreasing the categorization performance. Similarly, if there are too many C1 features per group, it is more likely that high values will be obtained in all the groups, thus reducing the informative value of the node. The number of features per group is therefore crucial for the feedforward recognition process and should provide a compromise between the two opposed effects described above.

Another factor that has proven crucial for successful categorization is the number of non-zero elements in the S2-C2 weight matrix, which can be considered equivalent to the sparseness of the matrix. There is evidence suggesting synaptic connectivity is sparse in feedforward cortical circuits and that firing patterns of cortical neurons exhibit sparse distributed representations, in which only a few of a large population of neurons are active (Quiroga et al. 2005, Murray and Kreutz-Delgado 2007, Karklin and Lewicki 2003, Olshausen 2003). Sparse coding strategies have proven to be essential to make efficient use of overcomplete representations, such as those found in V1, making it easier to find higher order correlations, increasing the signal-to-noise ratio and increasing the storage capacity of associative memories (Murray and Kreutz-Delgado 2007). Furthermore, they can improve pattern matching, since they lower the probability of false matches among elements of a pattern (Olshausen 2003).

The model results shown in Figures 5.12, 5.13, 5.14 and 5.15 indicate that sparse S2-C2 weight matrices, with $< 10\%$ of active connections, improve feedforward categorization. An example of one such sparse connectivity matrix is shown in Figure 4.11. As expected, the optimum number of non-zero elements is proportional to the S2 RF size. For S2 RF size=4x4, the optimum value of non-zero elements is one, while for higher S2 RF sizes the value lies between four and eight. As previously stated, sparse coding strategies account for this phenomenon, as more sparse S2-C2 connections make it less likely for two different objects to yield the same C2 response pattern (false positive), thus increasing selectivity. However, when the number of non-zero elements is too low, the distorted versions of the same object might be categorized as

different objects (false negative), leading to reduced invariance.

The graph in Figure 5.16 indicates that the S2 RF size also affects recognition performance but has different effects for each of the distorted test sets. The occluded test set works best with the smallest S2 RF size, probably because it better captures the non-occluded parts of the object, whereas the bigger RF sizes tend to include more occluded sections. Bigger RF sizes show a slight advantage when recognizing scaled objects, as the difference in size is less accentuated within large RFs, while it may lead to radically different smaller-sized features. Overall, it is clear that the best results are obtained by averaging over the four different S2 RF sizes, as one size's shortcomings are compensated for by another one's strengths.

Finally, a comparison between different models is shown in Figure 5.17, demonstrating that the proposed Bayesian network can achieve similar feedforward categorization results to the original HMAX model. Note that the comparison is only rigorously valid between the HMAX model and the Bayesian Belief Propagation (BBP) 3-level model, as these have equivalent numbers of layers, nodes and features per layer. The alternative BBP 3-level Yamane version was specifically modified, by reducing the pooling region and increasing the number of nodes of the top layers, to improve the categorization of the translated test set. Implementing the same modifications in the original HMAX model would, presumably, yield better results than the BBP version, in the same way that the original 3-level HMAX version produces better results than the 3-level BBP model.

The superior results of HMAX are, however, not surprising as it was specifically designed to perform feedforward categorization and employs more exact and sophisticated operations, namely the *max* and the Radial Basis function, than the BBP model. In fact, it is remarkable that the BBP model can achieve comparable categorization results using the local belief propagation operations, namely a weighted product operation for selectivity and a weighted sum operation for invariance. Crucially, using the same algorithm and structure, the BBP model also achieves recursive feedback modulation, which has been pinpointed as the major limitation of HMAX (Serre et al. 2005a).

With respect to the HTM-like model, as was previously noted, the Numenta Vision Toolkit was

used to compare the results, which allows for a maximum of 50 object categories. Although this means the test sets were different from those used for the rest of models, theoretically it confers an advantage to the HTM model as fewer categories facilitates the categorization task. Nonetheless, the relatively low performance of the model might be a consequence of not having enough training images per category, as the Numenta Vision Toolkit recommends having at least 20 training images per category. Furthermore, the internal structure of the HTM network is unknown, which means it is possible that this was not optimized for the type of images or categorization task employed, and alternative HTM networks could improve the results. Despite this, the results are intended to illustrate that it is not trivial that the task of feedforward categorization has been performed by belief propagation models that incorporate feedback functionality.

6.1.2 Feedback modulation and illusory contour completion

To test the effects of feedback in the network, the illusory contour completion paradigm was chosen. Experimental evidence strongly supports the involvement of high-level feedback in lower-level illusory contour development (Halgren et al. 2003, Lee and Nguyen 2001, Maertens et al. 2008). To try to reproduce this phenomenon, the setup was typically chosen to be a Kanizsa square as the input image to the network and the representation of a square fixed at some higher layer. The square representation was fed back from increasingly higher layers, ranging from C1 to S3. This was done in order to study the effects of feedback systematically and understand the particularities of each layer, although in last instance feedback should arise from the top layer after the Kanizsa image has been categorized as a square. Results are structured in the same way, providing a progressive account of the network's recurrent dynamics.

6.1.2.1 Feedback from C1 to S1

The first and most simple case (Figure 5.18) is that of feedback originating from the C1 layer. This example serves to clearly illustrate how bottom-up ($\lambda(S1)$) and top-down ($\pi(S1)$) evidence are multiplicatively combined in the S1 layer belief. Furthermore, it clarifies the correspondence between the probability distributions of the Bayesian nodes and the 2D graphical representations used throughout the Results chapter. Note that only the lower scale band of each layer is plotted,

as this contains the highest resolution representation. Layer S1 constitutes an exception where band 2 instead of 1 is shown, as the feedback weights from C1 layer band 1 fall predominantly in this band (see Figure 4.8). Although the information in higher bands might also contain relevant information and play an important role in perception, given the large scale of the model it was vital to focus the analysis on certain model responses in order to obtain a comprehensive understanding of them. Similarly, only the horizontal orientation state is shown, but due to the symmetry of the square, it is easy to interpret by extrapolation the response to the vertically oriented contours.

6.1.2.2 Feedback from S2 to S1

Figure 5.19 shows the setup and results of the case where the square feedback originates from the S2 layer. In contrast to the C1 feedback example, the C1 and S1 square reconstructions are now slightly more blurred, but still very clearly defined. It is possible to observe the gradual transformation from the Kanizsa figure to the square at all layers of the model, not only in the belief but in the λ and π messages. Note how it takes one time step for the square feedback to reach the C1 layer, but two time steps to reach S1. In both cases the illusory contour strength increases gradually over time, as depicted more clearly in Figure 5.20. This is a consequence of the modification introduced in the outward λ messages described in Section 4.5.3.3, which allows for the accumulation of belief responses, otherwise the response would remain identical after the second time step.

Note that C1 and S1 reconstructions show patterns of noise that repeat periodically. This is because the CPTs are derived from the prototype weight matrices, which are independent of position, and then particularized for the the set of nodes at each location. Even for flat parent distribution, the unbalanced CPTs lead to child nodes with non-flat distributions, as illustrated by the lower nodes of the example shown in Figure 3.8.

As stated before, it is necessary to focus on a subset of the model's responses, thus the results shown were obtained from the S2 nodes with RF size=4x4. However, Figure 5.21 demonstrates that the illusory contour develops for all S2 RF sizes, although the 4x4 size shows the most precise and least noisy reconstruction. Also, the figure illustrates how as the band size increases,

the image resolution decreases to a point where the square figure is unrecognizable, which justifies focusing on only the simulations on the lowest scale band. The original HMAX model was designed to process large natural images where the lower resolution of the higher bands might play a more important role.

6.1.2.3 Feedback from C2 to S2

The square reconstructions from feedback originating at layers C1 and S2 show a relatively good fit to the ideal square representation, even when using feedback weights equivalent to the feedforward weights. However, the loss of information between the S2 and C2 layer is much higher as it is mapping over 2000 nodes into 9 nodes. This is a general problem of modelling feedback connections in models that implement an invariance operation, such as the *max* function, which cannot be mapped backwards. For this reason, and because the feedforward weights proved to be inappropriate, a more systematic study was performed to elucidate what the key factors to obtain meaningful feedback from the C2 layer are.

Figures 5.23 and 5.24 show the results of testing three factors. The first one is the number of non-zero elements, or the inverse of sparseness, of the S2-C2 feedback weight matrix, which shows an almost linear, positive correlation with the ability of feedback to reconstruct an ideal C1 square representation. The second factor tested was the sampling parameters N_{C2} and K_{C2}, which, within the limited range of values tested due to the high computational cost, showed a very clear positive correlation with feedback's reconstruction capacity. The last factor studied was the number of non-zero elements in the S2-C2 feedforward weight matrix used to generate the C2 square representation, from where feedback originated. Although only two different values were tested, comparison between Figures 5.23 and 5.24 suggests that C2 representations generated using more non-zero elements in the feedforward weight matrices (less sparse) are better for feedback reconstruction.

It is important to note that the pixel-wise mean absolute error is not a perfect measure of the goodness of fit between the C1 feedback reconstruction and the ideal C1 square. For example, C1 reconstructions using a less sparse matrices tend to show a higher level of background noise or overall activity, which might lead to a lower error as they cover greater area of the ideal

square. Reconstructions using more sparse weight matrices may not cover as much area of the square but might be cleaner and more precise. Despite this, the mean absolute error provides an objective indicator of the goodness of fit between the reconstructions and can be used to guide the broad initial parameter search. This can be later refined for a smaller target parameter space using a more accurate measure.

Although a more exhaustive parameter search is required, the preliminary results obtained strongly suggest that asymmetric weight matrices are required: feedforward weights should be relatively sparse, leading to more selective higher-level representations; while feedback connection matrices and high-level representations require a higher density in order to increase the amount of information available to reconstruct the lower levels. This is consistent with evidence from cortex showing that feedforward connections tend to have sparse axonal bifurcation whereas backward connections have abundant axonal bifurcation. Furthermore, it agrees with the theoretical perspective that argues that a cell is likely to have few feedforward driving connections and many modulatory connections (Friston 2003).

Another parameter that is also likely to influence the feedback reconstruction is the number of features per group in the complex layers. Given the current implementation, where feedback to complex layers affects equally all the features belonging to a group, increasing the number of features per group will increase the overall amount of, still relatively diffuse, feedback. One important extension for the model would be to achieve heterogeneous feedback modulation of the features within a group. This can be done, for example, by allowing features to belong to different groups, such as in the HTM model (George and Hawkins 2009). The learning method in HTM automatically does this, whereas in the proposed model this could be achieved by finding correlations between features in different groups and then combining them into a single new group. A more comprehensive study of how this factor can aid the feedback disambiguation process is left as future work.

6.1.2.4 Feedback from C2 to S1

As discussed above, it is difficult to generate accurate feedback from the C2 square representation, so an alternative is to clamp the $\pi(S2)$ to the ideal S2 square representation as if the

feedback was really generated from C2. This allows observation of the temporal response in lower layers, including S2 itself. Figure 5.25 demonstrates that feedback originating in $\pi(S2)$ also leads to a very robust illusory contour completion effect in lower levels. Consistent with the hierarchical distance of the layers, the effect is now observed at t=2 in C1 and t=3 in S2. Figure 5.26 shows a more gradual development of the contour, compared to that shown in Figure 5.20 (feedback from $Bel(S2)$), due to the longer reciprocal interactions that now include the S2 layer.

An important aspect to study is how feedback from higher levels is refined as it interacts with bottom-up evidence. For this purpose, the model's response to different input images, given the same high level feedback, was compared in Figures 5.22 and 5.27. Both the S1 and C1 responses to the occluded Kanizsa square, blurred Kanizsa and to an empty input image, show only minor differences between them. The differences are observed around the real contours of the Kanizsa figures. For example, for the occluded Kanizsa figure, the vertical real contour of the occluding circle clearly stands out over the horizontal illusory contour. These small modulations of feedback become more apparent in Figure 5.27, specially when comparing the S1 response to the Kanizsa square versus the empty input image. These differences are shown in more detail in Figure 5.28.

This type of refinement would be expected to happen from complex to simple layers where the diffuse, low resolution feedback is sharpened based on existing low-level information that provides local cues to guide the disambiguation process (Halko et al. 2008, Lee 2003). Previous attempts to incorporate feedback connectivity into the HMAX model have encountered the same theoretical barrier, which basically deals with how to obtain spatial precision from invariant high-level abstract object representations (Dura-Bernal et al. 2010). The solution proposed previously was to implement a feedback disambiguation algorithm based on collinearity, co-orientation and good continuation principles, theoretically reproducing lateral connection functionality. In principle, this method illustrated well the feedforward-feedback interactions, however the algorithm was designed heuristically and worked exclusively with simple square-like figures.

The belief propagation algorithm in Bayesian networks is, theoretically, well-suited to implement these horizontal interactions. Pearl (1988), the first to formulate belief propagation in Bayesian networks, refers to them as *sideways* interactions (see Section 3.3.3). Although there are no explicit lateral connections, these are implemented implicitly by the bottom-up messages and top-down messages, both of which take into account evidence from nodes adjacent to the target. There are several possible reasons why, despite this, the results in Figure 5.25 (clamping of $\pi(S2)$ to square representation) don't show significant contextual lateral interactions and feedback disambiguation:

1. A number of approximations to the exact implementation of belief propagation have been made (see Section 4.6). These include sampling methods that limit the messages to relatively few samples which contain the highest information content. However, all the information that is lost due to the sampling and approximations might actually be required for precise feedback disambiguation. For example, features that present relatively low probabilities and could potentially be enhanced by feedback might be initially discarded during sampling.

2. All features belonging to the same group in complex layers are modulated equivalently by feedback. Features within groups contain the precise and high resolution information that could lead to belief refinement. As previously argued, allowing feedback to modulate features within a group disparately would lead to the enhancement of specific S1 spatial arrangements. This could be done by learning distinct weights for each feature or by allowing features to belong to different groups, both of which methods are implemented in the HTM mode (George and Hawkins 2009).

3. Loopy belief propagation might require more time steps to converge to a good approximation of the exact belief. Current simulations run for a limited number of time steps due to the high computational cost. Although beliefs tend to show a relatively high degree of convergence, it is possible that they are settling on local minima.

4. Beliefs are likely to evolve and be refined as a consequence of the hierarchical interactions

over time. The fact the both the bottom and top layers are clamped means that beliefs can only evolve freely along the intermediate layers, as the belief in peripheral layers will be dominated by the clamped representations. The present results suggest that if only the input image is clamped and beliefs allowed to evolve across the whole network, these *show greater contextual modulation through lateral interactions. This is illustrated in* Figure 5.31 and discussed below in Section 6.1.2.5.

It is also possible that the structure and parameters of the network, derived from the HMAX model and mimicking the ventral path, are not sufficient for the precise spatial refinement of feedback. Indeed the dorsal path, which has been shown to be tightly interlinked with the ventral path at many levels, may play a crucial role by providing spatial and motion related information which could guide the feedback disambiguation process (Fazl et al. 2009, Chikkerur et al. 2010, Grossberg et al. 2007). In this sense even for *static* images, such as those employed in this model, the continuous microsaccadic movements of the eye might be providing crucial information for perceptual completion processes (?). In this same line, George and Hawkins (2009) *demonstrated that simulating saccadic movements in the input image improved the feedback* reconstruction performance of the model. A more complete model of visual perception could therefore be accomplished by implementing a parallel interconnected Bayesian network that modelled the dorsal path and provided the additional information required.

Nonetheless, the results in Figure 5.28 demonstrate the ability of the model to feed back high-level information to lower levels, even in the absence of bottom-up input, consistent with evidence on mental imagery. Evidence has consistently shown that the regions and cortical representations of mental imagery are surprisingly similar to those of visual perception, suggesting both modalities share a common substrate (Ishai 2010). Slotnick et al. (2005) showed that visual mental imagery can evoke retinotopic activations in early visual regions, in agreement with generative modelling approaches. Recently, Reddy et al. (2010) obtained results suggesting the same patterns on neural activity generated during visual perception get reactivated during mental imagery, mediated by feedback connections from high-level object recognition layers.

The setup where feedback originates from $\pi(S2)$ was also used to explore the different belief

update methods during loopy belief propagation as shown in Figure 5.29. The different methods were explained in detail in Section 4.5.3. An initial first observation is that both the *upwards* method, used to obtain most of the results, and the *up-down* method provide similar results to that of the rigorously correct, but computationally expensive, *complete* update method. This argues for the validity of these methods, suggesting that computing the beliefs of layers where no new evidence information has arrived might be redundant and not provide any significant contribution to the final belief. However, it is likely that the differences between methods is accentuated as feedback originates from higher layers, leading to longer internal loops. Further and more systematic research is required to confirm the validity of these update methods.

Other interesting effects can be observed in Figure 5.29. To start with, the S1 belief seems to show an oscillatory effect where the illusory contour gets narrower and wider. It would be interesting to study in more detail whether the narrowing is a consequence of feedback disambiguation guided by the real contours of the input image or an epiphenomenon derived from some other cause. Also, the *up-down* method S1 belief shows the cleanest response, which is consistent with the fact that it is updating the layers in the expected sequence of propagation, minimizing the propagation of noisy information. According to this account, the illusory contour from the *upwards* method should be cleaner than that of the *complete* method, but is not. This might be a consequence of the asymmetry of the *upwards* method, which gives preference to the bottom-up evidence, as compared with the other two methods, where bottom-up and top-down evidence propagate at the same rate. Again, these are just speculative ideas based on the limited preliminary results obtained.

The results shown in Figure 5.30 demonstrate the ability of the model to feed back information from the C2 layer. Using the information from Figure 5.23, an S2-C2 weight matrix was chosen to try to obtain the best reconstruction possible, although this was still far from an ideal square representation. Nonetheless, the S1 and C1 layer still develop activity close to the illusory contour region.

6.1.2.5 Feedback from S3 to S1

Feedback emerging from the S3 layer was also able to generate the illusory contours in the lower layers as illustrated in Figure 5.31. This demonstrates that feedback is able to reconstruct the C2 square representation from the S3 layer information. In this case the *up-down* instead of the *upwards* belief update method was implemented as it produces a cleaner lower level response by reducing the accumulated noise from π messages. Importantly, this setup can be understood as a hypothetical scenario where the Kanizsa figure is correctly categorized as a square and due to some higher level mechanism, such as focused attention (Gilbert and Sigman 2007, Reynolds and Chelazzi 2004), only the square prototype is fed back, similar to a winner-take-all network.

Another important property, which was present in previous results but is more obvious here, is that the similarity of the internal representation of each layer to any fixed evidence (e.g. input image or high-level square representation) is proportional to the distance to the layer containing the evidence. In other words, lower layers show an internal representation close to the Kanizsa figure, whereas the representation in higher layers is closer to that of a square.This observation is consistent with evidence suggesting high-level activity generated by objects containing illusory contours is notably similar to the activity of complete objects (Stanley and Rubin 2003, Maertens et al. 2008, Sary et al. 2008). Furthermore, it is also in consonance with evidence showing the illusory contour response is weaker and only appears in a fraction of V1/V2 cells, in relation to that of real contours, and that V1 tends to show an even weaker illusory contour response than V2 (Lee 2003, Maertens et al. 2008, Seghier and Vuilleumier 2006, Halgren et al. 2003).

The temporal sequence of illusory contour formation in the model is also substantiated by experimental evidence showing that the LOC/IT region is the first to signal the appearance of the illusory contour, which then gradually spreads to lower regions (Murray et al. 2002, Halgren et al. 2003).

With respect to the mechanisms responsible for contour completion, Halko et al. (2008) affirmed in a recent review that illusory contours result from the interaction between high-level figural feedback and interpolation/extrapolation processes related to lateral connections. Figu-

ral feedback is clearly captured by the proposed model as the square representation is fed back exclusively from the top layer while the rest of the layers contain flat initial distributions. Interestingly, as shown in Figure 5.31, the feedback square reconstruction from higher layers, $\pi(S2)$, is significantly blurred and poorly defined. This is due to the previously described problems with feedback between the C2 and S2 layers. However, as time goes on, the square representation at both the S2 and C1 layers significantly improves. This suggests that feedback is interacting and being refined, or contextually modulated, by the lower layer activity (the Kanizsa figure). In other words, the square illusory contours emerge as a consequence of the reciprocal interaction over time of the Kanizsa figure inducers and the higher level abstract square representation, as suggested by experimental evidence (Halko et al. 2008).

The results in Figure 5.23 substantiate the contextual interaction claim by showing the more blurred C1 square representations that result from C2 feedback. This suggests that feedback by itself is insufficient to generate the more precise C1 square representations that emerge when using the Kaniza figure input image at the same time (Figure 5.31).

6.1.2.6 Object categorization with feedback

As described in Section 4.4, the categorization results are obtained by assuming the network has a singly-connected tree structure in order to avoid feedback modulation. Note that this refers strictly to the initial bottom-up pass where higher levels are assumed to contain flat distributions such that feedback would not provide any meaningful information. Evidence supports the theoretical view that the initial recognition process is indeed strictly a fast feedforward process with no feedback involvement (Masquelier and Thorpe 2007, Epshtein et al. 2008, Riesenhuber and Poggio 1999, Serre et al. 2007b). However, evidence also suggests that more cognitive priors, such as task-related attention, might have a fast effect on the local microcircuitry and modify the initial categorization performance (Lee and Nguyen 2001, Gilbert and Sigman 2007). For this reason, it was interesting to test whether similar categorization results could be obtained without any constraints on the network.

The preliminary results shown in Figure 5.32 confirm that the square object was correctly categorized using the *upwards* belief update method, which combines feedback information at each

step. Although feedback originates from empty high-level representations, it becomes non-flat as it is modulated by the conditional probability tables (CPTs) weights (see Section 4.5 for details), which explains why the resulting S3 distribution has more noise than the one with no feedback. Even the most extreme case which processed only the lowest band and implemented the *complete* belief update method (stronger feedback effects), situated the square prototype in fourth place, a surprisingly good result considering the limitations.

The Kanizsa square, which can be considered a strongly occluded square, was correctly categorized with no feedback and obtained significantly high positions for the *upward* update method with loopy feedback (first for the 4x4 S2 RF size and third for the averaged response). Even when reducing the number of bands to one and using the *complete* update method, the Kanizsa square still showed consistently good results. Overall, these results suggest that a similar categorization performance can be achieved by the model even when including the feedback loop during the initial bottom-up pass. However, further research is required to prove this hypothesis and to obtain a better understanding of the factors affecting feedforward categorization in loopy Bayesian networks.

The categorization of Kanizsa input images as squares is critical in order to simulate illusory contour completion without the need to clamp any high-level square representation. Instead the model should recognize the input Kanizsa figure as a square and feed back the corresponding information. The current categorization results using feedback do not provide an appropriate square representation, as the square state does not show the highest value or, if it does, the overall distribution is extremely noisy. This can be solved in the future by improving the feedforward categorization performance so that the Kanizsa figure elicits a clear square representation and by improving the feedback reconstruction from C2 to S2. This should allow to obtain an automatic illusory contour response just by feeding in the Kanizsa input image to the network. Current results using an idealized S3 square representation (Figure 5.31) are encouraging and support this claim as they manages to elicit the illusory contour in lower regions.

An interesting control test to perform would be to systematically rotate the Kanizsa pacmen by varying degrees. The categorization of the Kanizsa figure should be affected such that for

strong rotations the winner S3 prototype should no longer be a square object. Additionally, non-aligned inducers should prevent or reduce the strength of the illusory contours.

A different question to the one addressed in this section, is whether the feedback loop can improve categorization over time. Again, this can only be tested once the feedback reconstruction provided by higher layers is improved. As previously argued, the results shown in Figure 5.31 suggest that feedback may indeed improve categorization, based on how the C1 and S2 responses are gradually modulated towards a sharper square representation.

6.1.2.7 Feedback to S3

The example shown in Figure 5.33, despite depicting a very trivial problem, serves to illustrate the capacity of the model to simulate feedback effects, such as priming or expectation, which arise from areas outside the ventral pathway such as the prefrontal cortex, fusiform gyrus, posterior parietal cortex or the amygdala (Summerfield and Egner 2009, Bar et al. 2006, Grossberg et al. 2007, Sabatinelli et al. 2009, Gilbert and Sigman 2007). Furthermore, the model allows to simulate the activation of high-level object-selective regions due to mental imagery which has been suggested to be mediated by feedback connections from prefrontal cortex (Ishai 2010).

Importantly, these effects are accommodated as part of the Bayesian network parameters (S3 prior distribution), without the need to include any external artifacts. The example can also be interpreted as implementing feature attention (enhancing only states corresponding to animals in the S3 prior distribution) and could similarly implement spatial attention by defining a prior distribution that favours certain locations, specially when processing larger images with several objects. The Bayesian implementation of attention resembles that proposed by Chikkerur et al. (2010).

6.1.3 Benefits and limitations of Bayesian networks

Bayesian networks and belief propagation provide a rigorous mathematical framework, grounded in probability theory, that allows the feedforward and feedback interactions of a system to be modelled. One of its most attractive and arguably elegant features is its distributed implementation, wherein all the nodes have an homogeneous internal structure and carry out the same

operations. Specific functions can then be implemented by defining the appropriate structure and weights. It has been argued that this and other properties map well onto cortical connectivity and account for experimental evidence as described in Sections 3.1 and 3.2. Additionally, the model is well-suited for large-scale parallel implementation using asynchronous message-passing, such as that offered by multicore computers or hardware implementation (Jin et al. 2010, Neftci et al. 2010).

The model is, nonetheless, still a Bayesian network and thus cannot be considered biologically realistic. The model can only be argued to be realistic at a network or systems level of abstraction, which is closer to cognitive functionality than to biology. At this level of abstraction the network reproduces the same properties as the HMAX model, such as the hierarchical cortical structure and the tuning and invariance profiles of neurons at V1, V4 and inferotemporal (IT) cortex. This, of course, is still a strong simplification of the visual system. For example, direct reciprocal connections can be found between distant areas such as V1 and higher-level object-processing regions (Huang et al. 2007), which are not included in the model. Furthermore, the Gabor filters used to model V1 neurons RF and the distinction between simple and complex cells are an oversimplification of the wide spectrum of V1 neurons functionality (Ringach 2004). In addition, the response of neurons in higher cortical levels is still not well understood and thus, any attempt to model them is likely to be oversimplified and inaccurate (see Section 2.1.1 for further details).

Some of these effects could be accommodated by future versions of the model. For example, direct connections between the top and bottom layers of the model could be included by learning the appropriate weights, similar to one of the implemented version of HMAX (Serre et al. 2007b).

Regarding the complexity of neural responses, the proposed model has an advantage over HMAX in the sense that responses are modulated over time by the interaction between feed-forward and feedback connections. This accounts for extra-classical RF properties of neurons (Angelucci and Bullier 2003) and adds a large time-scale temporal dimension to the model responses (Kiebel et al. 2008) opposed to the static HMAX responses.

However, both models fail to capture any details at the neuronal level of description, such as the complex balance between excitatory and inhibitory connections or spike decoding including learning and adaptation mechanisms such as spike-timing dependent plasticity. Nevertheless, detailed biological implementations have been proposed both for the HMAX (Kouh and Poggio 2008, Yu et al. 2002, Knoblich et al. 2007) and the belief propagation operations (George and Hawkins 2009, Litvak and Ullman 2009, Steimer et al. 2009), which could theoretically allow the model to be implemented using spiking neurons. Importantly, given the large scale of the model, which spans three different cortical regions and has over two hundred thousand nodes, it seems reasonable to limit the level of detail until the principles tested have been shown to work.

Implementations of belief propagation, in general, assume each node corresponds to the computations performed by the microcircuits within a cortical column. Another interesting possibility is that single neurons act as nodes and approximate a simpler version of the algorithm, as proposed by Rao (2004) and Deneve (2008a). This approach has yielded some interesting results relating generative models to spike-time dependent plasticity (Nessler et al. 2009). Neural implementations of message-passing algorithms in graphical models are the current focus of research for several prestigious research centres, such as the Gatsby Institute in London and the Institute of Neuroinformatics in Zurich.

Importantly, the model might not be suitable for neural implementation in the present state due to the high redundancy in the information represented by the likelihood, belief and prior functions. A reformulation of the equations towards predictive coding approaches, wherein feedforward messages convey the prediction errors, could lead to more efficient implementations, in consonance with experimental evidence (Friston et al. 2006). Critically, predictive coding can be derived from belief propagation, which speaks for formal similarities between both approaches (Friston and Kiebel 2009, Kschischang et al. 2001, Yedidia et al. 2003).

The Bayesian network was designed based on the HMAX model, as this was a well-established model of the ventral path at the appropriate level of description. However, the HMAX structure might not be the ideal one for modelling visual perception using Bayesian networks, as it was designed exclusively for feedforward processing. For example, the Bayesian network

model could benefit from greater interactions among the lower level scale bands, which are currently processed in parallel. Furthermore, the HMAX design doesn't take into account the constraints of Bayesian networks, which may perform more efficiently using, for example, a smaller number of states per node.

Bayesian methods, such as the Expectation-Maximization algorithm, allow the optimum structure and parameters of a Bayesian network to be learned, given some data (Jordan and Weiss 2002, Lewicki and Sejnowski 1997, Murphy 2001). Although applying these methods from scratch to such large scale models might be computationally intractable, these can be used to shape the network given some initial structural constraints. The proposed model could potentially be formulated in a more generic format, similar to the HTM model (George and Hawkins 2009), which could then be particularized to specific scenarios with the aid of these Bayesian learning methods. The proposed model can be understood as a particularization of the more general model to the visual perception domain. However, the same generic model could be particularized to other similarly structured domains such as the auditory system.

For example, one of the main properties embodied by the generic model would be the simple and complex layer structure with complex layers grouping states in order to achieve invariance. Many of the potential generic principles have been described in Chapter 4, but a more detailed account and mathematical formulation of the generic framework is left as future work.

Several approximations and sampling methods, summarized in Section 4.6, have been implemented to deal with the large number of nodes and connections in the model. These offer solutions to the problem of multiplicatively combining a large number of discrete probability distributions with many states. Previous models have proposed performing calculations in the log domain to convert products into sums (Rao 2004, Litvak and Ullman 2009). Here I propose re-weighting distributions to establish a minimum value and sampling methods to keep only the highest values of the distributions with highest variance.

A further novelty of the model is to use the weighted sum model proposed by Das (2004) to approximate the CPT of nodes with multiple parents. Bayesian networks that try to model the visual cortex will irremediably require multiple parent interactions as this arise as a consequence

of overlapping receptive fields. Several methods have been proposed to approximate the exponential number of parameters of multiple-parent CPTs, the most common being the Noisy-OR gate (Pearl 1988, Diez 1993, Srinivas 1993, Onisko et al. 2001). This method however cannot be applied to variables that are not graded, such as those coding the different features as states of the variable. For this reason, the proposal by Das (2004), which has been justified from a geometrical perspective and is not constrained to graded variables, offers a valuable alternative.

The model also deals with loops in the network by implementing loopy belief propagation, a method that has only been proven to work empirically and constitutes an active field of research in itself (Murphy et al. 1999, Weiss 2000). The proposed model explores different belief updating methods and provides a comparison of the effects these have on the different layers over time. Additionally, to the best of my knowledge, this is the largest Bayesian network that implements loopy belief propagation and thus tests the limits and applicability of this approach. An alternative and potentially more efficient belief update method, which could be tested in future versions of the model, is asynchronous message-passing triggered by changes in the input to a node.

All of the above proposed methods are likely to be useful in the future for researchers modelling similar large-scale scenarios using Bayesian networks and belief propagation. However, it is difficult to evaluate the validity of these methods and their ability to approximate the exact beliefs of the network. The only way to obtain the exact marginal probabilities in networks with loops is to apply the junction-tree algorithm (Murphy et al. 1999), which would incur prohibitive computational costs. Thus, while these methods remain to be tested more systematically, the categorization performance and the feedback reconstruction capabilities of the model suggest the proposed methods point in the right direction. Furthermore, results from the setup where the square representation is fed from the top layer suggest lateral contextual interactions between the bottom-up input and feedback activity are present in the model.

On the other hand, the fact that these contextual lateral interactions are not clearly showing up in the results where feedback originates from S2 and C2, could suggest that the approximations and sampling methods used are discarding necessary information, as previously argued.

Alternatively, this could be a consequence of the model requiring interactions from the dorsal path in order to obtain spatial precision. A third option could be that invariance needs to be implemented in a different way, such as exploiting the inherent variability in the generative model reconstructions, instead of approximating the *max* operation in alternating layers. Small differences in the higher layer representation would lead to the repertoire of possible lower-level representations of a given object. However, this method has only been demonstrated for 28x28 pixel input images and using much smaller and constrained networks of binary nodes that replace the top layer with an associative memory (Hinton et al. 2006).

Current results support the inherent difficulty in developing a model that can achieve feedforward invariant object categorization, where position and scale information are lost, while at the same time achieving spatially precise feedback modulation. This limitation is present in previous similar models (Epshtein et al. 2008, Murray and Kreutz-Delgado 2007) and has only been partially solved by introducing temporal information (George and Hawkins 2009), spatial information from the *where* path (Chikkerur et al. 2009) or using heuristically defined algorithms for lateral interactions (Dura-Bernal et al. 2010).

Finally, it is important to point out that one of the main limitations of the model, despite the approximations and sampling methods implemented, is the considerable simulation time required. Using moderate sampling parameters, four time steps of the *upward* update method for layers S1 to S3 took over 60 hours. Depending on the parameter choice this value could vary between 5 hours and more than 100 hours. Several solutions are possible:

- Optimizing the MATLAB code by finding more efficient and faster implementations of the proposed algorithms.

- Implementing the model using a faster language such as C. The belief propagation algorithm for each node was implemented in C, which reduced the simulation time of each individual node to 50%, but the overall simulation time of the model was reduced to only 94%. This suggests much of the computation time is spent in routing the messages to the corresponding nodes.

- A real-time hardware implementation using large field-programmable gate arrays (FPGAs) or other parallel-computing systems such as the SpiNNaker (Jin et al. 2010). The proposed model is well suited to parallel distributed implementations such as those offered by hardware chips.

Reducing the simulation time would allow one to systematically explore the different parameters of the model and gain deeper insights into the approximations, sampling methods and results obtained by the simulations.

6.2 Comparison with experimental evidence

The proposed model is consistent with experimental evidence ranging from neuron physiology to anatomical data, and with several experimentally-grounded cortical theories. These are listed *below:*

- Widely accepted principles of object recognition in the ventral path, as supported by anatomical, biological, physiological and psychophysical data (Cadieu et al. 2007, Hung et al. 2005, Knoblich et al. 2007, Kouh and Poggio 2008, Masquelier et al. 2007, Riesenhuber and Poggio 1999, Serre et al. 2005a, 2007b, Serre and Riesenhuber 2004, Walther and Koch 2007, Yu et al. 2002). The Bayesian network reproduces the HMAX model operations and structure, which have been shown to capture these principles, and achieves invariant object categorization. See Section 2.1.2 for further details.

- The parallel, distributed and hierarchical architecture of the cortex is also reproduced by the inherent structure of Bayesian networks and belief propagation (Pearl 1988, Rao 2004, Lee and Mumford 2003, George and Hawkins 2009). Similarly, the homogeneous internal structure of cortical columns (the canonical microcircuit) is comparable to the homogeneous internal operations (belief propagation) of each Bayesian node (Friston and Kiebel 2009, George and Hawkins 2009, Steimer et al. 2009, Litvak and Ullman 2009). Possible cortical mappings of belief propagation and biologically plausible implementations have been thoroughly reviewed in Sections 3.4.1 and 3.4.3. Furthermore, an abundant body of evidence arguing for the, more general, Bayesian brain hypothesis was

presented in Section 3.2.

- The convergence of feedforward connections and the divergence of feedback connections (Friston 2003). The same pattern is found in the model where the number of parents of a node is always less than the number of children. The higher divergence of feedback connections accounts for contextual or extra-classical RF effects (Angelucci and Bullier 2003).

- The patchy axonal terminations of feedback connections and their functional specificity (Angelucci and Bullier 2003). Although feedback terminations have been commonly considered to be more diffuse and non-topographic (Friston 2003), recent findings show they have a very similar shape and density to those of feedforward connections, both for the V1-V2 (Anderson and Martin 2009) and V2-V4 (Anderson and Martin 2006) pathways. In the Bayesian network proposed, both the feedforward single CPTs and the feedback multiple parent CPTs are derived from the same weight matrices and thus exhibit similar connectivity properties. Nonetheless, further study of the S2-C2 weight matrix revealed that feedback requires a denser connectivity than feedforward processing. This would contradict Andersen's evidence, but be in agreement with the asymmetric connections theory and evidence showing the more sparse axonal bifurcation of feedforward versus feedback connections (Friston 2003).

- The illusory contour completion temporal response observed in the ventral system. As detailed in Section 2.3.1, the Kanizsa figure is represented as a complete figure in the higher levels and, as time progresses, an increasingly weaker representation can be observed in lower levels (Halgren et al. 2003, Maertens et al. 2008, Murray et al. 2002, Sary et al. 2008, Seghier and Vuilleumier 2006, Yoshino et al. 2006, Stanley and Rubin 2003, Lee and Nguyen 2001, Lee 2003). The model is shown to be consistent with the mechanisms proposed to be responsible for contour completion, namely, figural feedback and lateral interactions. Contextual lateral interactions were only observed when feedback originated from the top layer and was allowed to interact with bottom-up evidence forming recurrent loops across four layers. Possible reasons why these lateral interactions did

not clearly emerge when feedback originated from lower layers have been discussed in Section 6.1.2.

- Feedback effects that modulate the inferotemporal cortex arriving from the prefrontal cortex (object priming, expectation, etc.), the posterior parietal cortex (spatial attention), amygdala (emotional stimuli such as faces) and others (Bar et al. 2006, Grossberg et al. 2007, Summerfield and Egner 2009, Sabatinelli et al. 2009, Gilbert and Sigman 2007). These can be modelled by modifying the model S3 prior, $\pi(S3)$, to reflect the appropriate bias towards certain objects or locations.

- Feedback effects resulting exclusively from mental imagery with no bottom-up input. Evidence suggests that the same visual pathways are shared for visual perception and mental imagery resulting in similar cortical activations (Ishai 2010). Mental imagery is suggested to originate in prefrontal cortex, which feeds back to higher-level object recognition areas (Ishai 2010, Reddy et al. 2010). Further evidence has shown how mental imagery may lead to retinotopic activations in lower level visual regions (Slotnick et al. 2005). The proposed generative model can capture the feedback effects of mental imagery by modifying the S3 prior to simulate the mental image (feedback from prefrontal to inferotemporal cortex) and then allowing this to propagate to lower regions.

- The *active blackboard* hypothesis (Bullier 2001), high-resolution buffer (Lee 2003) and integrated model of visual perception (Lee et al. 1998), which argue for the parallel involvement of the ventral path in all stages of computation, rather than the classical feedforward cascade (see Section 2.2.2). The model updates all layers during each simulation time step reflecting the bottom-up and top-down interactions proposed in Figure 2.8.

- The Reverse Hierarchy Theory (Hochstein and Ahissar 2002), which states that explicit perception emerges first at the top level and then proceeds in a top-down fashion. The generative modelling approach implemented here is reminiscent of this theory, given that high level causes (objects) in the model unfold a series of lower level effects (features).

6.3 Comparison with previous models

The proposed model shares many structural and functional similarities with the Hierarchical Temporal Memory (HTM) model proposed by George and Hawkins (2009). They both employ *the belief propagation equations to approximate selectivity and invariance in alternating hierarchical layers*. The main difference is that the HTM nodes embody both the simple and complex features, which are called coincidence patterns and groups (Markov chains), respectively. The inclusion of a Markov chain within the node makes HTM qualitatively different from a Bayesian network. Consequently, belief propagation also becomes a qualitatively different algorithm that can be applied exclusively to HTM nodes. By combining simple and complex features within the same node, the authors avoid much of the complexity, and possibly benefits, inherent in a rigorous implementation of belief propagation, such as loops and multiple parents.

The proposed model implements the same feature grouping mechanism present in HTMs (except for the temporal correlation of Markov chains) by exploiting the weights of the CPTs between simple and complex layers. Figure A.1 in the Appendix Section A provides a schematic representation of an HTM network that implements the 3-level HMAX model (Serre et al. 2007c) used for this thesis. The HTM network is formulated using the original HTM notation (George and Hawkins 2009) combined with the original HMAX parameter notation (Serre et al. 2007c). The resulting HTM network can be compared to the Bayesian network that implements the same 3-level HMAX model (Figure 4.4) in order to obtain a better understanding of the differences between HTM and the proposed model.

The proposed model employs loopy belief propagation to perform approximate inference, similar to the HTM model (George and Hawkins 2009). Other models implementing approximate perceptual inference have employed message-passing algorithms derived from sampling methods (Hinton et al. 2006, Lee and Mumford 2003, Lewicki and Sejnowski 1997) or variational methods (Murray and Kreutz-Delgado 2007, Rao and Ballard 1999, Friston and Kiebel 2009).

The model by Epshtein et al. (2008) implements exact inference using belief propagation. However, it is employed over simplified networks with no loops and is qualitatively different from the proposed model in that nodes correspond to features and states to locations. The model

by Chikkerur et al. (2009) also implements exact inference on a Bayesian network but models exclusively high-level attention, such that the lower half of the network is non-Bayesian and strictly feedforward.

The type of input image used by the model is more complex and detailed than that of previous ones that were purely theoretical (Lee and Mumford 2003) or employed simplistic toy examples (Friston and Kiebel 2009, Lewicki and Sejnowski 1997, Hinton et al. 2006). Those with comparable input images fail to account for other properties that have been implemented by the proposed model, such as position and scale invariance (Rao and Ballard 1997, Murray and Kreutz-Delgado 2007, Chikkerur et al. 2009) or illusory contour completion (Hinton et al. 2006, Epshtein et al. 2008).

6.4 Future work

A number of potential improvements and extensions to the proposed model are listed below:

- Run simulations with Kanizsa figure controls that fully test the hypothesis that the model performs illusory contour completion. At the moment, the control data give an ambiguous answer to the model's performance.

- Perform a systematic analysis of the model parameters for both feedforward and feedback processing. Some of the key parameters to study are the number of features per group, the sparseness of the connectivity matrices and the sampling parameters.

- Learn heterogeneous feedback weights for features within a group and allow features to belong to different groups. This should improve the feedback disambiguation capacity and could lead to improved contextual modulation through lateral interactions.

- Improve the categorization of Kanizsa figures and the feedback C2-S2 reconstruction to allow automatic illusory contour completion without clamping feedback. This could also lead to an improvement of the categorization performance over time as a result of feedback modulation.

- Include adaptation mechanisms that could lead naturally to phenomena such as sensitivity

to temporal context and bistability (Mamassian and Goutcher 2005).

- *Include the lateral geniculate nucleus (LGN) as the bottom layer of the Bayesian network.* This would allow the corticothalamic feedback loop to be included within the same perceptual inference framework and compare model results with the detailed experimental data (Sillito et al. 2006).

- Increase the size of the input image to allow the simulation of multiple object detection, spatial attention and automatic attention-shifting (e.g. occluder vs. occluded object) (Walther and Koch 2007, Chikkerur et al. 2009). Additionally, natural images instead of silhouettes can be used. Thanks to the parametrized model implementation, no additional extension, apart from learning new weights, is required to test input images of different sizes and characteristics.

- Test the model using input images that change over time (movies). The hiearchical structure of the generative model should naturally lead to a hierarchy of time-scales similar to slow-feature analysis (Wiskott and Sejnowski 2002, Kiebel et al. 2008).

- Extend the model to include the *where* path containing spatial and motion information. This could be modelled as a parallel Bayesian network with cross-interactions with the *what* path at different levels.

- Formulate the model in a more generic way that can then be applied to different visual *scenarios or other domains, such as auditory perception. The generic formulation should* specify certain principles and constraints, describing how Bayesian networks and belief propagation can be applied to perceptual inference processes where selectivity and invariance are desired properties. The specific structure and parameters can then be partially learned using Bayesian learning methods.

- Real-time hardware implementation of the model using large parallel distributed systems, such as SpiNNaker (Jin et al. 2010).

6.5 Conclusions and summary of contributions

It is important to highlight that the claim made in this thesis is not that the visual cortex works exactly as a Bayesian network with belief propagation. However, the substantial body of evidence presented and the model results suggest that, at a functional and structural level of description, there exist significant similarities between the visual cortex and the proposed model.

Therefore, this thesis supports the notion that the role for feedback is not limited to attentional mechanisms, but provides a substrate for the exchange of information across the visual system leading to hierarchical perceptual inference. This thesis provides an explicit demonstration that Bayesian networks and belief propagation can be used as tools to model large-scale perceptual processes in the visual system. In this sense, it complements previous theoretical studies that argued for this approach (Lee 2003, Friston 2010) but did not provide an explicit implementation. At the same time, it complements small-scale biologically plausible implementations of belief propagation (Litvak and Ullman 2009, Steimer et al. 2009), by providing them with a large-scale functional model which they can attempt to reproduce. The proposed model can be used as a template to guide the design of large-scale biologically plausible implementations of belief propagation that capture the ventral path functionality.

A list of the contributions of this thesis is included below:

- A review and analysis of the experimental evidence, theories and computational models of the role of cortical high-level feedback in object perception, including the illusory and occluded contours.
- A review and analysis of the experimental evidence, theories and computational models suggesting the visual cortex can be understood in terms of a generative model, Bayesian networks and belief propagation. This includes a detailed comparison of existing functional models, biologically plausible implementations and possible cortical mappings.
- A comprehensive and mathematically rigorous explanation of belief propagation in Bayesian networks, including a novel, intuitive and illustrative example with numerical step-by-step demonstrations of the different types of evidence propagation.

- A Bayesian network that provides a probabilistic interpretation of the HMAX model and reproduces its structure; and an approximation to the selectivity and invariance operations of the HMAX model using the belief propagation algorithm over the proposed Bayesian network.

- An extension of the static feedforward HMAX model to include dynamic and recursive feedback based on the loopy belief propagation algorithm in the proposed Bayesian network.

- A particularization of the CPT learning method proposed by Das (2004) to the hierarchical object recognition domain. The method simplifies the generation of the CPT parameters for Bayesian networks where nodes have multiple parents.

- Solutions to the problems associated with the integration of information in large-scale Bayesian networks. These include sampling methods and the re-weighting of probability distributions to establish a minimum value.

- Simulation results and analysis demonstrating the model is consistent with anatomical, physiological and psychophysical data of the ventral path, including object categorization with invariance to occlusions, position and scale. Results also suggest categorization performance could improve over time by including the feedback loop, but further research is required to prove this hypothesis.

- Simulation results and analysis demonstrating the model is able to reproduce the phenomena of illusory contour formation, including the qualitative response pattern observed across layers, the temporal sequence of events and the mechanisms involved. An additional proof-of-concept example also demonstrates the model can account for higher-level feedback effects such as priming, attention and mental imagery. These results and the model implementation are shown to be consistent with a number of theoretical viewpoints such as the Reverse Hierarchy Theory, the *high-resolution buffer* hypothesis and the integrated model of visual perception.

- Analysis of the benefits and limitations of this model and, more generally, of using

Bayesian networks and belief propagation to model cortical object perception.

- A list of potential model extensions and improvements, and future lines of research in this field.

—

Appendix A

HMAX as a Hierarchical Temporal Memory network

A Hierarchical Temporal Memory (HTM) network can be specified mathematically as a generative model and is defined by the following parameters:

- HTM nodes $N^{L,i}$, where L = level of the hierarchy, and i = index of the node within that level.
- Each node contains a set of patterns $c_1, ..., c_M$ and a set of groups/Markov chains $g_1, ..., g_N$, each of which is defined over a subset of the coincidence patterns in that node.
- Connectivity between child and parent nodes which defines the structure bottom-up messages λ and top-down messages π of each node during belief propagation.

Figure A.1 provides a schematic representation of how an HTM network could implement the 3-level HMAX model (Serre et al. 2007c). The diagram defines all the above parameters for an HTM network that captures the structure and connectivity of the 3-level HMAX implementation. The parameters of the HTM network are described as a function of the parameters of the HMAX model, using the same notation as in Table 4.2. To summarize:

- Each HTM node corresponds to all the HMAX complex units at a specific location, and all of its afferent simple units.
- The HTM groups correspond to each of the features coded by the HMAX complex units at that location.

- The HTM patterns correspond to the features coded HMAX simple units. We assume simple units with a different relative location to the complex unit, represent a different HTM pattern.

Thus, HTM nodes embody both the simple and complex features, which are called coincidence patterns and groups (Markov chains), respectively. The inclusion of the groups within the node makes HTM qualitatively different from a Bayesian network. Consequently, belief propagation also becomes a qualitatively different algorithm that can be applied exclusively to HTM nodes. By combining simple and complex features within the same node, the authors avoid much of the complexity inherent in a rigorous implementation of belief propagation, such as loops and multiple parents. The resulting HTM network can be compared to the Bayesian network that *implements the same 3-level HMAX model (Figure 4.4) in order to obtain a better understand*ing of the differences between HTM and the proposed model.

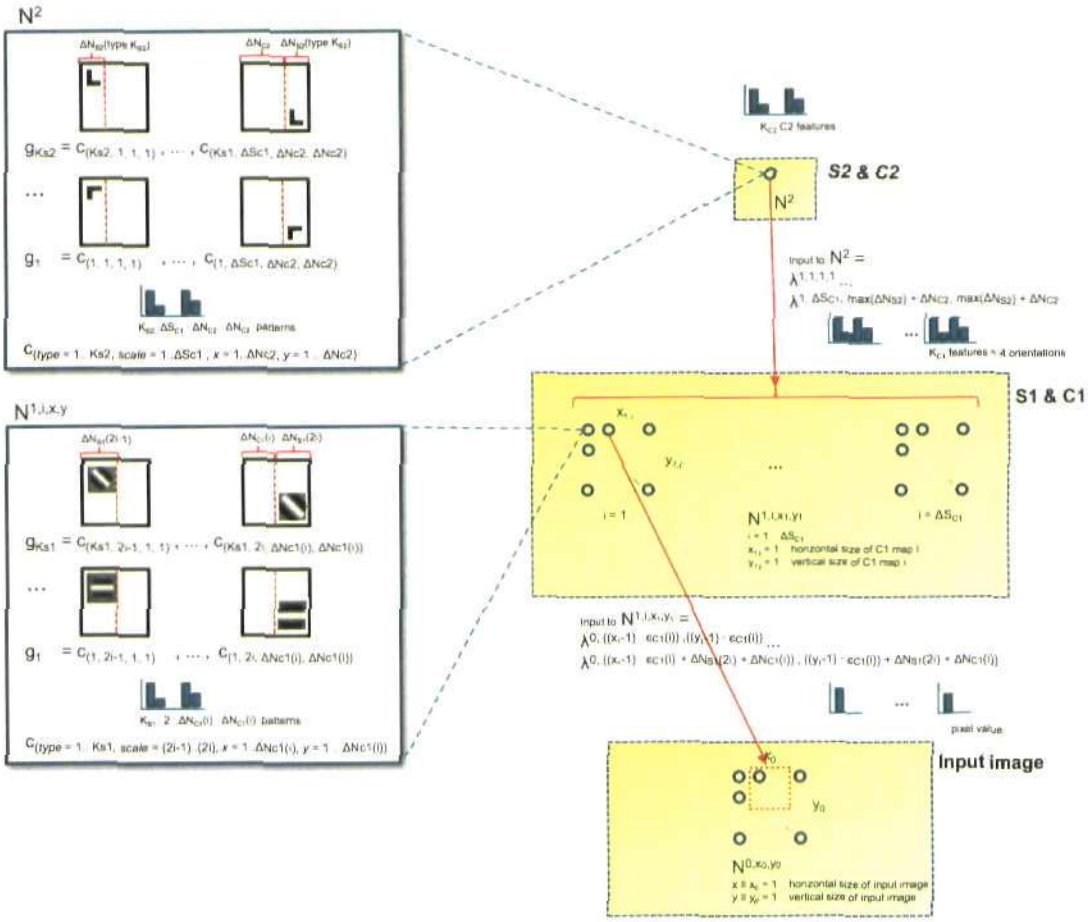


Figure A.1: Schematic representation of how an Hierarchical Temporal Memory (HTM) network could implement the 3-level HMAX model (Serre et al. 2007c). HTM nodes embody both the simple and complex features, which are called coincidence patterns and groups (Markov chains), respectively. The HTM network is formulated using the original HTM notation (George and Hawkins 2009) combined with the original HMAX parameter notation (Serre et al. 2007c). See text for details.

Glossary.

$\lambda(x)$	Likelihood function, which combines all bottom-up evidence of node X
$\lambda_C(x)$	Bottom-up message from node C to node X
$\pi(x)$	Prior function, which combines all top-down evidence of node X
$\pi_X(u)$	Top-down message from node U to node X
$Bel(x)$	Belief, or posterior probability of node X
ART	Adaptive Resonance Theory
CPT	Conditional probability table, equivalent to the connectivity matrix between Bayesian nodes
EEG	Electroencephalography
fMRI	functional magentic resonance imaging
IT	Inferotemporal cortex
LGN	Lateral geniculate nucleus
LOC	Lateral occipital complex
MEG	magnetoencephalography
MST	Medial superior temporal cortex
MT	Middle temporal cortex
PP	Posterior parietal cortex
RHT	Reverse Hierarch Theory
STDP	Spike-time dependent plasticity
V1	Primary visual cortex
V2	Secondary visual cortex

List of references.

Ahissar, M., Nahum, M., Nelken, I. & Hochstein, S. (2009), 'Reverse hierarchies and sensory learning', *Philosophical Transactions of the Royal Society B: Biological Sciences* **364**(1515), 285–299.

Alink, A., Schwiedrzik, C. M., Kohler, A., Singer, W. & Muckli, L. (2010), 'Stimulus predictability reduces responses in primary visual cortex', *J. Neurosci.* **30**(8), 2960–2966.

Anderson, J. C. & Martin, K. A. (2006), 'Synaptic connection from cortical area v4 to v2 in macaque monkey', *The Journal of Comparative Neurology* **495**(6), 709–721.

Anderson, J. C. & Martin, K. A. C. (2009), 'The synaptic connections between cortical areas v1 and v2 in macaque monkey', *J. Neurosci.* **29**(36), 11283–11293.

Andolina, I. M., Jones, H. E., Wang, W. & Sillito, A. M. (2007), 'Corticothalamic feedback enhances stimulus response precision in the visual system', *Proceedings of the National Academy of Sciences* **104**(5), 1685–1690.

Angelucci, A. & Bullier, J. (2003), 'Reaching beyond the classical receptive field of vi neurons: horizontal or feedback axons?', *Journal of Physiology-Paris* **97**(2-3), 141–154.

Angelucci, A., Levitt, J. B., Walton, E. J. S., Hupe, J. M., Bullier, J. & Lund, J. S. (2002), 'Circuits for local and global signal integration in primary visual cortex', *Journal of Neuroscience* **22**(19), 8633–8646.

Bar, M., Kassam, K. S., Ghuman, A. S., Boshyan, J., Schmid, A. M., Dale, A. M., Hamlainen, M. S., Marinkovic, K., Schacter, D. L., Rosen, B. R. & Halgren, E. (2006), 'Top-down facilitation of visual recognition', *Proceedings of the National Academy of Sciences of the United States of America* **103**(2), 449–454.

Bayerl, P. & Neumann, H. (2004), 'Disambiguating visual motion through contextual feedback modulation', *Neural Computation* **16**(10), 2041–2066.

Beal, M. (2003), Variational Algorithms for Approximate Bayesian Inference, Phd thesis, Gatsby Computational Neuroscience Unit, University College London.

Bhatt, R., Carpenter, G. & Grossberg, S. (2007), Texture segregation by visual cortex: Perceptual grouping, attention, and learning, Technical Report Technical Report CAS/CNS-TR-2006-007, Boston University.

Bishop, C. (1995), *Neural networks for pattern recognition*, Oxford University Press.

Bolz, J. & Gilbert, C. D. (1986), 'Generation of end-inhibition in the visual cortex via interlaminar connections', *Nature* **320**(6060), 362–365.

Bullier, J. (2001), 'Integrated model of visual processing', *Brain Research Reviews* **36**(2-3), 96–107.

Cadieu, C., Kouh, M., Pasupathy, A., Connor, C. E., Riesenhuber, M. & Poggio, T. (2007), 'A model of v4 shape selectivity and invariance', *Journal of Neurophysiology* **98**(3), 1733–1750.

Carandini, M., Demb, J. B., Mante, V., Tolhurst, D. J., Dan, Y., Olshausen, B. A., Gallant, J. L. & Rust, N. C. (2005), 'Do we know what the early visual system does?', *Journal of Neuroscience* **25**(46), 10577–10597.

Carpenter, G. & Grossberg, S. (1987), 'A massively parallel architecture for a self-organizing neural pattern recognition machine', *Comput. Vision Graph. Image Process.* **37**(1), 54–115.

Carpenter, G. & Grossberg, S. (1998), 'Adaptive resonance theory (art)', pp. 79–82.

Chater, Courville, Kording, K. P., Michel, Shultz, Steyvers, Tenenbaum, J., Yuille, A. & Griffiths (2006), 'Special issue: Probabilistic models of cognition', *Trends in Cognitive Science* **10**(7).

Chikkerur, S., Serre, T., Tan, C. & Poggio, T. (2009), 'An integrated model of visual attention using shape-based features', *Massachusetts Institute of Technology, Cambridge, MA.* **CBCL paper 278 /MIT-CSAIL-TR 2009-029**.

Chikkerur, S., Serre, T., Tan, C. & Poggio, T. (2010), 'What and where: A bayesian inference theory of attention', *Vision Research* **In Press, Corrected Proof**.

Das, B. (2004), 'Generating conditional probabilities for bayesian networks: Easing the knowledge acquisition problem', *CoRR* **cs.AI/0411034**.

Dayan, P., Hinton, G. E., Neal, R. M. & Zemel, R. S. (1995), 'The helmholtz machine', *Neural Computation* **7**(5), 889–904.

De Meyer, K. & Spratling, M. W. (2009), 'A model of non-linear interactions between cortical top-down and horizontal connections explains the attentional gating of collinear facilitation', *Vision Research* **49**(5), 553–568.

Dean, T. (2006), Scalable inference in hierarchical generative models, *in* 'In the Proceedings of the Ninth International Symposium on Artificial Intelligence and Mathematics'.

Deco, G. & Rolls, E. T. (2004), 'A neurodynamical cortical model of visual attention and invariant object recognition', *Vision Research* **44**(6), 621–642.

Deneve, S. (2005), Bayesian inference in spiking neurons, *in* L. Saul, Y. Weiss & L. Bottou, eds, 'Advances in Neural Information Processing Systems', Vol. 17, MIT Press, Cambridge, MA, pp. 353–360.

Deneve, S. (2008*a*), 'Bayesian spiking neurons i: Inference', *Neural Computation* **20**(1), 91–117.

Deneve, S. (2008*b*), 'Bayesian spiking neurons ii: Learning', *Neural Computation* **20**(1), 118–145.

Desimone, R. & Duncan, J. (1995), 'Neural mechanisms of selective visual attention', *Annual Review of Neuroscience* **18**(1), 193–222.

Diez, F. (1993), 'Parameter adjustment in bayes networks: The generalized noisy or-gate'.

Dillenburger, B. (2005), Perception and processing of illusory contours, PhD thesis, Eberhard-Karls-University of Tuebingen.

Dura-Bernal, S., Wennekers, T. & Denham, S. (2010), The role of feedback in a hierarchical model of object perception, *in* 'Proceedings of BICS 2010 - Brain Inspired Cognitive Systems 14-16 July 2010, Madrid, Spain.'.

Epshtein, B., Lifshitz, I. & Ullman, S. (2008), 'Image interpretation by a single bottom-up top-down cycle', *Proceedings of the National Academy of Sciences* **105**(38), 14298–14303.

Fang, F., Kersten, D. & Murray, S. O. (2008), 'Perceptual grouping and inverse fmri activity patterns in human visual cortex', *Journal of Vision* **8**(7), 1–9.

Fazl, A., Grossberg, S. & Mingolla, E. (2009), 'View-invariant object category learning, recognition, and search: How spatial and object attention are coordinated using surface-based attentional shrouds', *Cognitive Psychology* **58**(1), 1–48.

Felleman, D. & Van Essen, D. (1991), 'Distributed hierarchical processing in primate cerebral cortex', *Cerebral Cortex* **1**(1), 1–47.

Fregnac, Y. (2010), Cortical hierarchies, networks and daisies, *in* 'The 2010 CapoCaccia Cognitive Neuromorphic Engineering Workshop'.

Friston, K. (2003), 'Learning and inference in the brain', *Neural Networks* **16**(9), 1325–1352.

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

Friston, K. (2010), 'The free-energy principle: a unified brain theory?', *Nat Rev Neurosci* **11**(2), 127–138.

Friston, K. & Kiebel, S. (2009), 'Cortical circuits for perceptual inference', *Neural Networks* **22**(8), 1093–1104.

Friston, K., Kilner, J. & Harrison, L. (2006), 'A free energy principle for the brain', *Journal of Physiology-Paris* **100**(1-3), 70–87.

Friston, K. & Stephan, K. (2007), 'Free-energy and the brain', *Synthese* **159**(3), 417–458.

Fukushima, K. (1988), 'Neocognitron: A hierarchical neural network capable of visual pattern recognition', *Neural Networks* **1**(2), 119–130.

Galuske, R. A. W., Schmidt, K. E., Goebel, R., Lomber, S. G. & Payne, B. R. (2002), 'The role of feedback in shaping neural representations in cat visual cortex', *Proceedings of the National Academy of Sciences of the United States of America* **99**(26), 17083–17088.

Garrido, M., Kilner, J., Kiebel, S. & Friston, K. (2007), 'Evoked brain responses are generated by feedback loops', *Proceedings of the National Academy of Sciences* **104**(52), 20961–20966.

Gawne, T. J. & Martin, J. M. (2002), 'Responses of primate visual cortical v4 neurons to simultaneously presented stimuli', *Journal of Neurophysiology* **88**(3), 1128–1135.

Geisler, W. S. & Kersten, D. (2002), 'Illusions, perception and bayes', *Nature Neuroscience* **5**(6), 508–510.

George, D. & Hawkins, J. (2009), 'Towards a mathematical theory of cortical micro-circuits', *PLoS Comput Biol* **5**(10), e1000532.

Giese, M. A. & Poggio, T. (2003), 'Neural mechanisms for the recognition of biological movements', *Nature Reviews Neuroscience* **4**(3), 179–192.

Gilbert, C. D. & Sigman, M. (2007), 'Brain states: Top-down influences in sensory processing', *Neuron* **54**(5), 677–696.

Girard, P., Hupe, J. M. & Bullier, J. (2001), 'Feedforward and feedback connections between areas v1 and v2 of the monkey have similar rapid conduction velocities', *Journal of Neurophysiology* **85**(3), 1328–1331.

Gollisch, T. & Meister, M. (2010), 'Eye smarter than scientists believed: Neural computations in circuits of the retina', *Neuron* **65**(2), 150–164.

Grossberg, S. (2003), 'How does the cerebral cortex work? development, learning, attention, and 3d vision by laminar circuits of visual cortex.', *Behavioral and Cognitive Neuroscience Reviews* **2**, 47–76.

Grossberg, S., Cisek, P., Drew, T. & Kalaska, J. F. (2007), Towards a unified theory of neocortex: laminar cortical circuits for vision and cognition, *in* 'Progress in Brain Research', Vol. Volume 165, Elsevier, pp. 79–104.

Grossberg, S., Mingolla, E. & Ross, W. D. (1997), 'Visual brain and visual perception: how does the cortex do perceptual grouping?', *Trends in Neurosciences* **20**(3), 106–111.

Grossberg, S. & Raizada, R. D. S. (2000), 'Contrast-sensitive perceptual grouping and object-based attention in the laminar circuits of primary visual cortex', *Vision Research* **40**(10-12), 1413–1432.

Guo, K., Robertson, R. G., Pulgarin, M., Nevado, A., Panzeri, S., Thiele, A. & Young, M. P. (2007), 'Spatio-temporal prediction and inference by v1 neurons', *European Journal of Neuroscience* **26**(4), 1045–1054.

Halgren, E., Mendola, J., Chong, C. D. R. & Dale, A. M. (2003), 'Cortical activation to illusory shapes as measured with magnetoencephalography', *NeuroImage* **18**(4), 1001–1009.

Halko, M. A., Mingolla, E. & Somers, D. C. (2008), 'Multiple mechanisms of illusory contour perception', *Journal of Vision* **8**(11), 1–17.

Harrison, L. M., Stephan, K. E., Rees, G. & Friston, K. J. (2007), 'Extra-classical receptive field effects measured in striate cortex with fmri', *Neuroimage* **34**(3), 1199–1208.

Hayworth, K. J. & Biederman, I. (2006), 'Neural evidence for intermediate representations in object recognition', *Vision Research* **46**(23), 4024–4031.

Hegde, J., Fang, F., Murray, S. O. & Kersten, D. (2008), 'Preferential responses to occluded objects in the human visual cortex', *Journal of Vision* **8**(4), 1–16.

Hegde, J. & Van Essen, D. C. (2007), 'A comparative study of shape representation in macaque visual areas v2 and v4', *Cereb. Cortex* **17**(5), 1100–1116.

Heitger, F., von der Heydt, R., Peterhans, E., Rosenthaler, L. & KÃijbler, O. (1998), 'Simulation of neural contour mechanisms: representing anomalous contours', *Image and Vision Computing* **16**(6-7), 407–421.

Hinton, G. E., Dayan, P., Frey, B. J. & Neal, R. M. (1995), 'The wake-sleep algorithm for unsupervised neural networks', *Science* **268**(5214), 1158–1161.

Hinton, G. E., Osindero, S. & Teh, Y. (2006), 'A fast learning algorithm for deep belief nets', *Neural Comput* **18**(7), 1527–54.

Hochstein, S. & Ahissar, M. (2002), 'View from the top: Hierarchies and reverse hierarchies in the visual system', *Neuron* **36**(5), 791–804.

Hoffman, K. L. & Logothetis, N. K. (2009), 'Cortical mechanisms of sensory learning and object recognition', *Philosophical Transactions of the Royal Society B: Biological Sciences* **364**(1515), 321–329.

Huang, J. Y., Wang, C. & Dreher, B. (2007), 'The effects of reversible inactivation of postero-temporal visual cortex on neuronal activities in cat's area 17', *Brain Research* **1138**, 111–128.

Hubel, D. H. & Wiesel, T. N. (1965), 'Receptive fields and functional architecture in two nonstriate visual areas (18 and 19) of the cat', *Journal of Neurophysiology* **28**, 229–289.

Hulme, Oliver, J. & Zeki (2007), 'The sightless view: Neural correlates of occluded objects', *Cerebral Cortex* **17**(5), 1197–1205.

Hung, C. P., Kreiman, G., Poggio, T. & Dicarlo, J. J. (2005), 'Fast readout of object identity from macaque inferior temporal cortex', *Science* **310**(5749), 863–6.

Hupe, J. M., James, A. C., Girard, P., Lomber, S. G., Payne, B. R. & Bullier, J. (2001), 'Feedback connections act on the early part of the responses in monkey visual cortex', *Journal of Neurophysiology* **85**(1), 134–145.

Huxlin, K. R., Saunders, R. C., Marchionini, D., Pham, H.-A. & Merigan, W. H. (2000), 'Perceptual deficits after lesions of inferotemporal cortex in macaques', *Cereb. Cortex* **10**(7), 671–683.

Ishai, A. (2010), 'Seeing with the mind's eye: top-down, bottom-up, and conscious awareness', *F1000 Biol Reports* **2**(34).

Jehee, J. F. M. & Ballard, D. H. (2009), 'Predictive feedback can account for biphasic responses in the lateral geniculate nucleus', *PLoS Computational Biology* **5**(5), e1000373. doi:10.1371/journal.pcbi.1000373.

Jin, X., Luj, M., Khan, M. M., Plana, L. A., Rast, A. D., Welbourne, S. R. & Furber, S. B. (2010), Algorithm for mapping multilayer bp networks onto the spinnaker neuromorphic hardware, *in* 'Proceedings of the 2010 Ninth International Symposium on Parallel and Distributed Computing', IEEE Computer Society.

Johnson, J. S. & Olshausen, B. A. (2005), 'The recognition of partially visible natural objects in the presence and absence of their occluders', *Vision Res* **45**(25-26), 3262–3276.

Jones, J. P. & Palmer, L. A. (1987), 'An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex', *Journal of Neurophysiology* **58**(6), 1233–1258.

Jordan, M. I. & Weiss, Y. (2002), Graphical models: probabilistic inference, *in* M. Arbib, ed., 'Handbook of Neural Networks and Brain Theory', 2nd edition edn, MIT Press.

Karklin, Y. & Lewicki, M. S. (2003), 'Learning higher-order structures in natural images', *Network: Computation in Neural Systems* **14**(3), 483–499.

Kellman, P. J. (2003), 'Interpolation processes in the visual perception of objects', *Neural Networks* **16**(5-6), 915–923.

Kellman, P. J., Guttman, S. E. & Wickens, T. D. (2003), Geometric and neural models of object perception, *in* T. F. Shipley & P. J. Kellman, eds, 'From fragments to objects: Segmentation and Grouping in Vision', *Elsevier Science, New York, p. 183 âĂŞ 245*.

Kersten, D., Mamassian, P. & Yuille, A. (2004), 'Object perception as bayesian inference', *Annual Review of Psychology* **55**, 271–304.

Kiani, R., Esteky, H., Mirpour, K. & Tanaka, K. (2007), 'Object category structure in response patterns of neuronal population in monkey inferior temporal cortex', *J Neurophysiol* **97**(6), 4296–4309.

Kiebel, S. J., Daunizeau, J. & Friston, K. J. (2008), 'A hierarchy of time-scales and the brain', *PLOS Computational Biology* **4**(11), e1000209. doi:10.1371/journal.pcbi.1000209.

Knill, D. C. & Richards, W., eds (1996), *Perception as Bayesian inference*, Cambridge University Press.

Knoblich, U., Bouvrie, J. V. & Poggio, T. (2007), Biophysical models of neural computation: max and tuning circuits, *in* N. Zhong, J. Liu, Y. Yao, J.-L. Wu, S. Lu & K. Li, eds, 'Web Intelligence Meets Brain Informatics', Vol. 4845 of *Lecture Notes in Computer Science*, Springer, Beijing, pp. 164–189.

Kokkonen, Kouivusalo, LAine, Jolma & Varis (2005), A method for defining conditional probability tables with link strength parameters for a bayesian network, *in* A. Zerger & R. Argent, eds, 'MODSIM05 International Congress on Modelling and Simulation Advances and Applications for Management and Decision-Making.', Melbourne.

Kording, K. P. & Wolpert, D. M. (2004), 'Bayesian integration in sensorimotor learning', *Nature* **427**(6971), 244–247.

Kouh, M. & Poggio, T. (2008), 'A canonical neural circuit for cortical nonlinear operations', *Neural Computation* **20**(6), 1427–1451.

Kschischang, F. R., Frey, B. J. & Loeliger, H. A. (2001), 'Factor graphs and the sum-product algorithm', *IEEE Transactions on Information Theory* **47**(2), 498–519.

Lampl, I., Ferster, D., Poggio, T. & Riesenhuber, M. (2004), 'Intracellular measurements of spatial integration and the max operation in complex cells of the cat primary visual cortex', *Journal of Neurophysiology* **92**(5), 2704–2713.

Lanyon, L. & Denham, S. (2009), 'Modelling attention in individual cells leads to a system with realistic saccade behaviours', *Cognitive Neurodynamics* **3**(3), 223–242.

Lanyon, L. J. & Denham, S. L. (2004), 'A biased competition computational model of spatial and object-based attention mediating active visual search', *Neurocomputing* **58-60**, 655–662.

Lee, T. & Nguyen, M. (2001), 'Dynamics of subjective contour formation in the early visual cortex', *Proceedings of the National Academy of Sciences* **98**(4), 1907–1911.

Lee, T. S. (2003), 'Computations in the early visual cortex', *Journal of Physiology-Paris* **97**, 121–139.

Lee, T. S. & Mumford, D. (2003), 'Hierarchical bayesian inference in the visual cortex', *Journal of the Optical Society of America A: Optics, Image Science and Vision* **20**(7), 1434–1448.

Lee, T. S., Mumford, D., Romero, R. & Lamme, V. A. F. (1998), 'The role of the primary visual cortex in higher level vision', *Vision Research* **38**(15-16), 2429–2454.

Leopold, D. A. & Logothetis, N. K. (1996), 'Activity changes in early visual cortex reflect monkeys' percepts during binocular rivalry', *Nature* **379**(6565), 549–553.

Lewicki, M. S. & Sejnowski, T. J. (1997), Bayesian unsupervised learning of higher order structure, *in* M. C. Mozer, M. I. Jordan & T. Petsche, eds, 'Advances in Neural Information Processing Systems', Vol. 9, The MIT Press, p. 529.

Li, Z. (2001), 'Computational design and nonlinear dynamics of a recurrent network model of the primary visual cortex', *Neural Computation* **13**(8), 1749–1780.

Litvak, S. & Ullman, S. (2009), 'Cortical circuitry implementing graphical models', *Neural Computation* p. In press.

Logothetis, N. K., Pauls, J., BÃijlthoff, H. H. & Poggio, T. (1994), 'View-dependent object recognition by monkeys', *Current Biology* **4**(5), 401–414.

Logothetis, N. K., Pauls, J. & Poggio, T. (1995), 'Shape representation in the inferior temporal cortex of monkeys', *Current Biology* **5**(5), 552–563.

Ma, W. J., Beck, J. M., Latham, P. E. & Pouget, A. (2006), 'Bayesian inference with probabilistic population codes', *Nature Neuroscience* **9**(11), 1432–1438.

Macknik, S. & Martinez-Conde, S. (2007), 'The role of feedback in visual masking and visual processing', *Advances in Cognitive Psychology* **3**(1), 125–152.

Macknik, S. & Martinez-Conde, S. (2009), The role of feedback in visual attention and awareness, *in* Gazzaniga, ed., 'The Cognitive Neurosciences', MIT Press, pp. 1165–1180.

Maertens, M., Pollmann, S., Hanke, M., Mildner, T. & MÃűller, H. E. (2008), 'Retinotopic activation in response to subjective contours in primary visual cortex', *Frontiers in Human Neuroscience* **2**(2), doi:10.3389/neuro.09.002.2008.

Mamassian, P. & Goutcher, R. (2005), 'Temporal dynamics in bistable perception', *Journal of Vision* **5**(4), 361–375.

Marr, D. (1982), *Vision*, Freeman, San Francisco, California.

Masquelier, T., Serre, T., Thorpe, S. & Poggio, T. (2007), Learning complex cell invariance from natural videos: a plausibility proof, Technical report, Massachusetts Institute of Technology.

Masquelier, T. & Thorpe, S. (2010), Learning to recognize objects using waves of spikes and spike timing-dependent plasticity, *in* 'IEEE IJCNN'.

Masquelier, T. & Thorpe, S. J. (2007), 'Unsupervised learning of visual features through spike timing dependent plasticity', *PLoS Computational Biolology* **3**(2), e31.

Mathias, F., Niko, W. & Laurenz, W. (2008), Invariant object recognition with slow feature analysis, *in* 'Proceedings of the 18th international conference on Artificial Neural Networks, Part I', Springer-Verlag, Prague, Czech Republic.

Montero, V. M. (1991), 'A quantitative study of synaptic contacts on interneurons and relay cells of the cat lateral geniculate nucleus', *Experimental Brain Research* **86**(2), 257–270.

Moore, C. & Engel, S. A. (2001), 'Neural response to perception of volume in the lateral occipital complex', *Neuron* **29**(1), 277–286.

Mumford, D. (1996), Pattern theory: a unifying perspective, *in* D. Knill & W. Richards, eds, 'Perception as Bayesian Inference', Cambridge Univ. Press, pp. 25–62.

Murphy, K. (2001), 'An introduction to graphical models'.

Murphy, K. (2002), Dynamic Bayesian networks: representation, inference and learning, Phd thesis, UC Berkeley, Computer Science Division.

Murphy, K., Weiss, Y. & Jordan, M. (1999), Loopy belief propagation for approximate inference: An empirical study, *in* 'Proceedings of the 15th Annual Conference on Uncertainty in Artificial Intelligence (UAI-99)', Morgan Kaufmann, San Francisco, CA, pp. 467–47.

Murray, J. & Kreutz-Delgado, K. (2007), 'Visual recognition and inference using dynamic overcomplete sparse learning', *Neural Computation* **19**(9), 2301–2352.

Murray, M. M., Imber, M. L., Javitt, D. C. & Foxe, J. J. (2006), 'Boundary completion is automatic and dissociable from shape discrimination', *J. Neurosci.* **26**(46), 12043–12054.

Murray, M. M., Wylie, G. R., Higgins, B. A., Javitt, D. C., Schroeder, C. E. & Foxe, J. J. (2002), 'The spatiotemporal dynamics of illusory contour processing: Combined high-density electrical mapping, source analysis, and functional magnetic resonance imaging', *Journal of Neuroscience* **22**(12), 5055–5073.

Murray, S. O., Schrater, P. & Kersten, D. (2004), 'Perceptual grouping and the interactions between visual cortical areas', *Neural Networks* **17**(5-6), 695–705.

Nassi, J. J. & Callaway, E. M. (2009), 'Parallel processing strategies of the primate visual system', *Nature Reviews Neuroscience* **10**(5), 360–372.

Neftci, E., Chicca, E., Cook, M., Indiveri, G. & Douglas, R. (2010), State-dependent sensory processing in networks of vlsi spiking neurons, *in* 'Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on'.

Nessler, B., Pfeiffer, M. & Maass, W. (2009), 'Stdp enables spiking neurons to detect hidden causes of their inputs', *In Proc. of NIPS 2009: Advances in Neural Information Processing Systems* **22**.

Neumann, H. & Mingolla, E. (2001), Computational neural models of spatial integration in perceptual grouping., *in* T.F.Shipley & P. Kellman, eds, 'From Fragments to Objects: Grouping and Segmentation in Vision', Elsevier, Amsterdam, pp. 353–400.

Olshausen, B. (2003), Principles of image representation in visual cortex, *in* J. W. L.M. Chalupa, ed., 'The Visual Neurosciences', MIT Press, pp. 1603–15.

Olshausen, B. & Field, D. (2005), 'How close are we to understanding v1?', *Neural Computation* **17**(8), 1665–1699.

Onisko, A., Druzdzel, M. J. & Wasyluk, H. (2001), 'Learning bayesian network parameters from small data sets: application of noisy-or gates', *International Journal of Approximate Reasoning* **27**(2), 165–182.

Pasupathy, A. & Connor, C. (2002), 'Population coding of shape in area v4', *Nat Neurosci* **5**(12), 1332–1338.

Pasupathy, A. & Connor, C. E. (2001), 'Shape representation in area v4: Position-specific tuning for boundary conformation', *J Neurophysiol* **86**(5), 2505–2519.

Pearl, J. (1988), *Probabilistic reasoning in intelligent systems: networks of plausible inference*, Morgan.

Peters, A., Payne, B. R. & Budd, J. (1994), 'A numerical analysis of the geniculocortical input to striate cortex in the monkey', *Cerebral Cortex* **4**(3), 215–229.

Pouget, A., Dayan, P. & Zemel, R. S. (2003), 'Inference and computation with population codes', *Annual Review of Neuroscience* **26**, 381–410.

Quiroga, Q., Reddy, L., Kreiman, G., Koch, C. & Fried, I. (2005), 'Invariant visual representation by single neurons in the human brain', *Nature* **435**(7045), 1102–1107.

Raizada, R. D. S. & Grossberg, S. (2001), 'Context-sensitive binding by the laminar circuits of v1 and v2: A unified model of perceptual grouping, attention, and orientation contrast', *Visual Cognition* **8**(3), 431 – 466.

Raizada, R. D. S. & Grossberg, S. (2003), 'Towards a theory of the laminar architecture of cerebral cortex: computational clues from the visual system', *Cerebral Cortex* **13**(1), 100–113.

Ramsden, B. M., Hung, C. P. & Roe, A. W. (2001), 'Real and illusory contour processing in area v1 of the primate: a cortical balancing act', *Cereb. Cortex* **11**(7), 648–665.

Rao, R. P. (2005), Hierarchical bayesian inference in networks of spiking neurons, *in* 'Advances in NIPS', Vol. 17, Vancouver, British Columbia, Canada.

Rao, R. P. N. (1999), 'An optimal estimation approach to visual perception and learning', *Vision Research* **39**(11), 1963–1989.

Rao, R. P. N. (2004), 'Bayesian computation in recurrent neural circuits', *Neural Computation* **16**(1), 1–38.

Rao, R. P. N. (2006), Neural models of bayesian belief propagation, *in* K. Doya, ed., 'Bayesian brain: Probabilistic approaches to neural coding', MIT Press, pp. 239–268.

Rao, R. P. N. & Ballard, D. (2005), Probabilistic models of attention based on iconic representations and predictive coding, *in* L. Itti, G. Rees & J. Tsotsos, eds, 'Neurobiology of Attention', Academic Press.

Rao, R. P. N. & Ballard, D. H. (1997), 'Dynamic model of visual recognition predicts neural response properties in the visual cortex', *Neural Computation* **9**(4), 721–763.

Rao, R. P. N. & Ballard, D. H. (1999), 'Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects', *Nature Neuroscience* **2**(1), 79–87.

Rauschenberger, R., Liu, T., Slotnick, S. D. & Yantis, S. (2006), 'Temporally unfolding neural representation of pictorial occlusion', *Psychological Science* **17**(4), 358–364.

Reddy, L., Tsuchiya, N. & Serre, T. (2010), 'Reading the mind's eye: Decoding category information during mental imagery', *NeuroImage* **50**(2), 818–825.

Reynolds, J. H. & Chelazzi, L. (2004), 'Attentional modulation of visual processing', *Annual Review of Neuroscience* **27**(1), 611–647.

Riesenhuber, M. & Poggio, T. (1999), 'Hierarchical models of object recognition in cortex', *Nature Neuroscience* **2**(11), 1019–25.

Ringach, D. L. (2004), 'Mapping receptive fields in primary visual cortex', *The Journal of Physiology* **558**(3), 717–728.

Roelfsema, P. (2006), 'Cortical algorithms for perceptual grouping', *Annual review of neuroscience* **29**, 203–227.

Rolls, E. T. & Milward, T. (2000), 'A model of invariant object recognition in the visual system: Learning rules, activation functions, lateral inhibition, and information-based performance measures', *Neural Computation* **12**(11), 2547–2572.

Rust, N. C. & DiCarlo, J. J. (2010), 'Selectivity and tolerance ("invariance") both increase as visual information propagates from cortical area v4 to it', *J. Neurosci.* **30**(39), 12978–12995.

Sabatinelli, D., Lang, P. J., Bradley, M. M., Costa, V. D. & Keil, A. (2009), 'The timing of emotional discrimination in human amygdala and ventral visual cortex', *J. Neurosci.* **29**(47), 14864–14868.

Sary, G., Koteles, K., Kaposvari, P., Lenti, L., Csifcsak, G., Franko, E., Benedek, G. & Tompa, T. (2008), 'The representation of kanizsa illusory contours in the monkey inferior temporal cortex', *European Journal of Neuroscience* **28**, 2137–2146.

Sasaki, Y., Nanez, J. E. & Watanabe, T. (2010), 'Advances in visual perceptual learning and plasticity', *Nat Rev Neurosci* **11**(1), 53–60.

Schwartz, O., Hsu, A. & Dayan, P. (2007), 'Space and time in visual context', *Nat Rev Neurosci* **8**(7), 522–535.

Seghier, M. L. & Vuilleumier, P. (2006), 'Functional neuroimaging findings on the human perception of illusory contours', *Neuroscience and Biobehavioral Reviews* **30**(5), 595–612.

Serre, T. (2006), Learning a dictionary of shape-components in visual cortex: comparison with neurons, humans and machines, Phd thesis, Massachusetts Institute of Technology.

Serre, T., Kouh, M., Cadieu, C., Knoblich, U., Kreiman, G. & Poggio, T. (2005), 'A theory of object recognition: Computations and circuits in the feedforward path of the ventral stream in primate visual cortex.', *Massachusetts Institute of Technology, Cambridge, MA* **CBCL Paper 259/AI Memo 2005-036**.

Serre, T., Kreiman, G., Kouh, M., Cadieu, C., Knoblich, U. & Poggio, T. (2007), A quantitative theory of immediate visual recognition, *in* 'Progress In Brain Research, Computational Neuroscience: Theoretical Insights into Brain Function', Vol. 165C, Elsevier, pp. 33–56.

Serre, T., Oliva, A. & Poggio, T. (2007), 'A feedforward architecture accounts for rapid categorization', *Proceedings of the National Academy of Sciences* **104**(15), 6424–6429.

Serre, T. & Riesenhuber, M. (2004), 'Realistic modeling of simple and complex cell tuning in the hmax model, and implications for invariant object recognition in cortex', *Massachusetts Institute of Technology, Cambridge, MA.* **CBCL Paper 239/AI Memo 2004-017**.

Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M. & Poggio, T. (2007), 'Robust object recognition with cortex-like mechanisms', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **29**(3), 411–426.

Serre, T., Wolf, L. & Poggio, T. (2005), Object recognition with features inspired by visual cortex, *in* 'Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on', Vol. 2, pp. 994–1000 vol. 2.

Sillito, A. M., Cudeiro, J. & Jones, H. E. (2006), 'Always returning: feedback and sensory processing in visual cortex and thalamus', *Trends in Neurosciences* **29**(6), 307–316.

Sillito, A. M. & Jones, H. E. (2008), 'The role of the thalamic reticular nucleus in visual processing', *Thalamus Related Systems* **4**(01), 1–12.

Slotnick, S. D., Thompson, W. L. & Kosslyn, S. M. (2005), 'Visual mental imagery induces retinotopically organized activation of early visual areas', *Cerebral Cortex* **15**(10), 1570–1583.

Soltani, A. & Wang, X.-J. (2010), 'Synaptic computation underlying probabilistic inference', *Nat Neurosci* **13**(1), 112–119.

Spratling, M. (2008*a*), 'Reconciling predictive coding and biased competition models of cortical function', *Frontiers in Computational Neuroscience* **2**(4), 1–8.

Spratling, M. W. (2008*b*), 'Predictive coding as a model of biased competition in visual attention', *Vision Research* **48**(12), 1391–1408.

Spratling, M. W. (2010), 'Predictive coding as a model of response properties in cortical area v1', *J. Neurosci.* **30**(9), 3531–3543.

Srinivas, S. (1993), A generalization of the noisy-or model, *in* 'Proceedings of the Ninth Conference on Uncertainty in Artificial Intelligence', Washington, DC.

Stanley, D. A. & Rubin, N. (2003), 'fmri activation in response to illusory contours and salient regions in the human lateral occipital complex', *Neuron* **37**(2), 323–331.

Steimer, A., Maass, W. & Douglas, R. (2009), 'Belief propagation in networks of spiking neurons', *Neural Computation* **21**(9), 2502–2523.

Sterzer, P., Haynes, J. D. & Rees, G. (2006), 'Primary visual cortex activation on the path of apparent motion is mediated by feedback from hmt+/v5', *Neuroimage* **32**(3), 1308–1316.

Summerfield, C. & Egner, T. (2009), 'Expectation (and attention) in visual cognition', *Trends in Cognitive Sciences* **13**(9), 403–409.

Tanaka, K. (1997), 'Mechanisms of visual object recognition: monkey and human studies', *Current Opinion in Neurobiology* **7**(4), 523–529.

Thomson, A. & Lamy, C. (2007), 'Functional maps of neocortical local circuitry', *Frontiers in Neuroscience* **1**(1), 19–42.

Tiesinga, P. H. & Buia, C. I. (2009), 'Spatial attention in area v4 is mediated by circuits in primary visual cortex', *Neural Networks* **22**(8), 1039–1054.

Tsunoda, K., Yamane, Y., Nishizaki, M. & Tanifuji, M. (2001), 'Complex objects are represented in macaque inferotemporal cortex by the combination of feature columns', *Nat Neurosci* **4**(8), 832–838.

Ullman, S. (2007), 'Object recognition and segmentation by a fragment-based hierarchy', *Trends in Cognitive Sciences* **11**(2), 58–64.

Van Essen, D. C. & Gallant, J. L. (1994), 'Neural mechanisms of form and motion processing in the primate visual system', *Neuron* **13**(1), 1–10.

Wallis, G. & Rolls, E. T. (1997), 'Invariant face and object recognition in the visual system', *Progress in Neurobiology* **51**(2), 167–194.

Walther, D. B. & Koch, C. (2007), 'Attention in hierarchical models of object recognition', *Progress in Brain Research* **165**, 57–78.

Weigelt, S., Singer, W. & Muckli, L. (2007), 'Separate cortical stages in amodal completion revealed by functional magnetic resonance adaptation', *BMC Neuroscience* **8**(1), 70.

Weiss, Y. (1997), 'Belief propagation and revision in network with loops', *MIT A.I. Memo No .1616. C.B.C.L. Paper No. 155* .

Weiss, Y. (2000), 'Correctness of local probability propagation in graphical models with loops', *Neural Computation* **12**(1), 1–41.

Weiss, Y. & Adelson, E. H. (1998), Slow and smooth: a bayesian theory for the combination of local motionsignals in human vision, Technical Report AIM-1624, MIT.

Williams, M. A., Baker, C. I., Op de Beeck, H. P., Mok Shim, W., Dang, S., Triantafyllou, C. & Kanwisher, N. (2008), 'Feedback of visual object information to foveal retinotopic cortex', *Nature Neuroscience* **11**(12), 1439–1445.

Wilson, H. R. (2003), 'Computational evidence for a rivalry hierarchy in vision', *Proceedings of the National Academy of Sciences of the United States of America* **100**(24), 14499–14503.

Winn, J. & Bishop, C. M. (2005), 'Variational message passing', *J. Mach. Learn. Res.* **6**, 661–694.

Wiskott, L. & Sejnowski, T. J. (2002), 'Slow feature analysis: Unsupervised learning of invariances', *Neural Computation* **14**(4), 715–770.

Wu, S. & Amari, S. (2001), Neural implementation of bayesian inference in population codes, *in* 'Advances in Neural Information Processing Systems 14 (NIPS*2001).'.

Yamane, Y., Tsunoda, K., Matsumoto, M., Phillips, A. N. & Tanifuji, M. (2006), 'Representation of the spatial relationship among object parts by neurons in macaque inferotemporal cortex', *J Neurophysiol* **96**(6), 3147–3156.

Yedidia, J., Freeman, W. & Weiss, Y. (2003), Understanding belief propagation and its generalizations, *in* 'Exploring artificial intelligence in the new millennium', Morgan Kaufmann Publishers Inc., San Francisco, CA, pp. 239–269.

Yoshida, W., Seymour, B., Friston, K. J. & Dolan, R. J. (2010), 'Neural mechanisms of belief inference during cooperative games', *J. Neurosci.* **30**(32), 10744–10751.

Yoshino, A., Kawamoto, M., Yoshida, T., Kobayashi, N., Shigemura, J., Takahashi, Y. & Nomura, S. (2006), 'Activation time course of responses to illusory contours and salient region: A high-density electrical mapping comparison', *Brain Res* .

Yu, A. J., Giese, M. A. & Poggio, T. A. (2002), 'Biophysiologically plausible implementations of the maximum operation', *Neural Computation* **14**(12), 2857–2881.

Yuille, A. & Kersten, D. (2006), 'Vision as bayesian inference: analysis by synthesis?', *Trends in Cognitive Sciences* **10**(7), 301–308.

Zemel, R., Huys, Q., Natarajan, R. & Dayan., P. (2004), Probabilistic computation in spiking populations, *in* 'Advances in Neural Information Processing Systems 17'.