Massively Parallel Spiking Neural Circuits: Encoding, Decoding and Functional Identification

Yiyin Zhou

Submitted in partial fulfillment of the
requirements for the degree
of Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2015

ABSTRACT

Massively Parallel Spiking Neural Circuits:
Encoding, Decoding and Functional Identification

Yiyin Zhou

This thesis presents a class of *massively* parallel spiking neural circuit architectures in which neurons are modeled by dendritic stimulus processors cascaded with spike generators. We investigate how visual stimuli can be represented by the spike times generated by the massively parallel neural circuits, how the spike times can be used to reconstruct and process visual stimuli, and the conditions when visual stimuli can be faithfully represented/reconstructed. Functional identification of the massively parallel neural circuits from spike times and its evaluation are also investigated. Together, this thesis offers a comprehensive analytic framework of massively parallel spiking neural circuit architectures arising in the study of early visual systems.

In encoding, modeling of visual stimuli in reproducing kernel Hilbert spaces is presented, recognizing the importance of studying visual encoding in a rigorous mathematical framework. For massively parallel neural circuits with biophysical spike generators, I/O characterization of the biophysical spike generators becomes possible by introducing phase response curve manifolds for the biophysical spike generators. I/O characterization of the entire neural circuit can then be interpreted as generalized sampling in the Hilbert space. Multi-component dendritic stimulus processors are introduced to model visual encoding in stereoscopic color vision. It is also shown that encoding of visual stimuli by an ensemble of complex cells has the complexity of Volterra dendritic stimulus processors.

Based on the I/O characterization, reconstruction algorithms are derived to decode, from spike times, visual stimuli encoded by these massively parallel neural circuits. Decoding problems are first formulated as spline interpolation problems. Conditions on faithful reconstruction are presented, allowing the probe of information content carried by the spikes.

Algorithms are developed to qualify the decoding in massively parallel settings. For stereoscopic color visual stimuli, demixing of individual channels from an unlabeled set of spike trains is demonstrated. For encoding with complex cells, decoding problems are formulated as rank minimization problems. It is shown that the decoding algorithm does not suffer from the curse of dimensionality and thereby allows for a visual representation using biologically realistic neural resources.

The study of visual stimuli encoding and decoding enables the functional identification of massively parallel neural circuits. The duality between decoding and functional identification suggests that algorithms for functional identification of the projection of dendritic stimulus processors onto the space of input stimuli can be formulated similarly to the decoding algorithms. Functional identification of dendritic stimulus processors of neurons carrying stereoscopic color information as well as that of energy processing in complex cells is demonstrated. Furthermore, this duality also inspires a novel method to evaluate the quality of functional identification of massively parallel spiking neural circuits. By reconstructing novel stimuli using identified circuit parameters, the evaluation of the entire identified circuit is reduced to intuitive comparisons in stimulus space.

The use of biophysical spike generators advances a methodology in the study of intrinsic noise sources in neurons and their effects on stimulus representation and on precision of functional identification. These effects are investigated using a class of nonlinear neural circuits consisting of both feedforward and feedback Volterra dendritic stimulus processors and biophysical spike generators. It is shown that encoding with neural circuits with intrinsic noise sources can be interpreted as generalized sampling with noisy measurements. Effects of noise on decoding and functional identification are derived theoretically and were systematically investigated by extensive simulations.

Finally, the massively parallel neural circuit architectures are shown to enable the implementation of identity preserving transformations in the spike domain using a switching matrix regulating the connection between encoding and decoding. Two realizations of the architectures are developed, and extensive examples using continuous visual streams are provided. Implications of this result on the problem of invariant object recognition in the spike domain are discussed.

Table of Contents

List of Figures

Acknowledgments

I would like to thank my advisor, Aurel A. Lazar, for his long-term support. I was introduced to this field to understand the brain by Prof. Lazar in his exciting lectures. This genuine excitement about creating new ideas was always inspiring during my Ph.D. study and will continue to encourage me. I thank Prof. Lazar for finding time for the countless discussions, including those 15-minute quick brainstorming and those 4-hour marathons, regardless of day or night, weekdays or weekends. His attitude and commitment to research had deeply influenced me to push myself to reach the same level.

I thank the members of my dissertation committee, Professors Ning Qian, Paul Sajda, Qi Wang and John Wright, for their time and for providing insightful feedbacks and suggestions on my work. I am also grateful to Professor Nima Mesgarani for being on my thesis proposal committee.

I am fortunate to have worked with the former and current members of the Bionet lab, including Lev Givon, Anmo Kim, Wenze Li, Eftychios Pnevmatikakis, Konstantinos Psychas, Yevgeniy Slutskiy, Robert Turetsky, Nikul Ukani and Chung-Heng Yeh. In particular, I would like to thank Lev for excellent maintenance of our computing devices, a necessary condition for the work in this thesis. I thank Eftychios, Yevgeniy and Nikul, who I have worked closely with, for all the discussions, debates, ideas that pushed forward our projects. My gratitude also goes to Chung-Heng for his helps on many things.

I am greatly indebted to my parents, Jiayi Zhou and Qinfang Wu, for their patient and unconditional support, without which I could not have gone so far. I am grateful to my loving wife, Yanxia Fang, for being by my side during my Ph.D. study, and for her unconditional love and the understanding of me working at nights and during weekends.

To my parents

Jiayi Zhou and Qinfang Wu

and

to my wife

Yanxia Fang

Chapter 1

Introduction

1.1 Motivation

Vision is *massive*.

The visual environment contains a massive amount of information that is crucial for the survival of animals. Most diurnal animals (including humans) rely heavily on visual information during their lifetimes. Vision guides them in navigating the world; vision enables them to quickly recognize and locate threats; vision helps them to find and capture food.

Vision seems to be such a trivial task. As one opens their eyes, it takes no time to see the surrounding environment. What happens internally, however, is not as simple as it seems for the visual system to represent and process the massive amount of information that naturally arises in the visual environment. What is required is a parallel circuit with a complexity matching that of the visual environment in order to facilitate rapid visual representation and processing. As a result, the visual systems evolved to be massive.

The human retina, for example, consists of more than 100 million neurons, of which 1.2 to 1.5 million are retinal ganglion cells (RGCs) that comprise the parallel outputs to the cortex. A further expansion takes place in the primary visual cortex (V1), where visual scenes are represented/processed by 150 million neurons [189]. Even in the tiny brain of the fruit fly *Drosophila melanogaster* that is comprised of ~$100,000$ neurons, 80% of its neurons are involved in vision [77].

The parallelism in the visual system applies not only to the number of neurons that fulfill the same task, but also to the number of different circuits that each performs a different computation/representation. For example, the vertebrate retina has more than 20 different types of outputs [33]. Presumably, each type of circuit carries a part of the visual information, and together, they form a more complete representation of the visual world.

It is natural, then, to ask:

- *How is visual information represented by massively parallel neural circuits?*

Experiments in the past 60 years have hinted at the role of individual neurons in representing visual scenes. These date back to the work by Kuffler on the response of RGCs [93] as well as the Nobel prize-winning work by Hubel and Wiesel on characterizing the response of neurons in V1 [79]. Recent advances in theoretical and experimental techniques have revealed the structure of their processing in much finer detail. What has been shown is that neurons in different parts of the early visual system carry diverse information about the visual scene. The response of some neurons can be related to visual stimuli linearly, while the response of others show nonlinear relation [22]. Some neurons carry color information [34], and some encode visual scenes observed by both eyes [57]. The relationship between input stimuli and the response of individual neurons in the early visual systems is quite well understood. Understanding of what an individual neuron does is not sufficient to understand how an entire visual scene is represented, however. What is missing is the ability to combine these findings and show that the response of an ensemble of neurons do represent the visual scene.

This creates a significant difficulty in understanding the massively parallel neural circuits in the early visual systems by employing experiments, as even the largest multi-electrode array currently can only record from approximately 1,000 neurons simultaneously. This number is only a fraction of the neurons in the retina, not to mention those in the visual cortex.

In an inspiring experiment, however, the recorded responses of approximately 200 neurons in the cat's Lateral Geniculate Nucleus (LGN) were used to reconstruct a visual scene

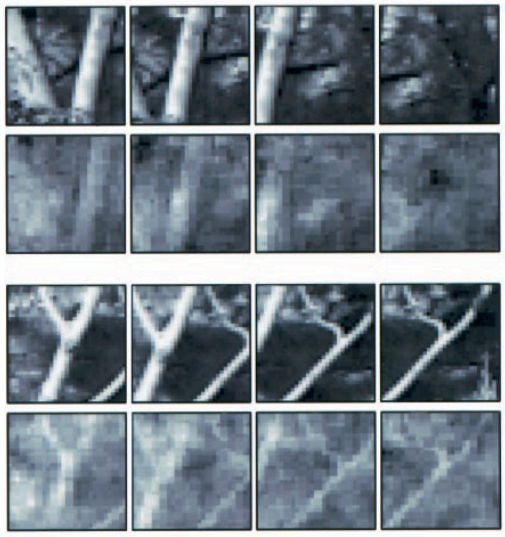


Figure 1.1: Reconstruction of visual stimuli from an ensemble of cat LGN neurons. The first and the third rows show a few original frames in the visual stimuli presented to the cat. The second and fourth rows show the corresponding frame in the video reconstructed from responses of the LGN neurons. (Modified from [177] with permission, ©1999 Society for Neuroscience)

viewed by a cat [177]. The decoded visual stimuli, although being very coarse and having a very small screen size, bore a resemblance of the original visual scene. A few example frames are shown in Figure 1.1.

The experiment suggested a promising path to build an "oscilloscope" to probe the information content carried by an ensemble of neurons by reconstructing the stimulus encoded. The question of whether the visual stimuli can be reconstructed from neural responses is not about decoding, but it is actually a question about encoding. The decodability of neural responses suggests how much information content in the original visual stimuli can be represented by the neural code, or how much information is retained after neural representation/processing.

In this sense, many of the current approaches aimed at reproducing neural responses fall short in that the underlying encoding schemes are empirical. It is not clear whether the spikes generated by a massively parallel neural circuit faithfully represent the visual stimuli

encoded, what stimulus variability (bandwidth) the neuronal response can encode, or what circuit complexity is required to do so.

In this dissertation, we aim to expand on the previous experimental findings, and base the study of visual encoding on a formal framework in which we show conditions when visual stimuli can be faithfully reconstructed, and therefore conditions when visual stimuli can be faithfully represented. We also seek to show that this framework is compatible with visual stimuli with large screen size, and that the representation takes the form of time of spikes generated by massively parallel neural circuits.

Spikes, or action potentials, are the natural language of nervous systems. The nature of spikes, their representational and processing power, have been investigated and debated over many decades, yet they remain unresolved. Many neural codes have been proposed under either rate coding, in which the average spiking activity carries information, or temporal coding, in which the precise timing of spikes is used for representation and processing.

In this dissertation, we focus on the representation of stimuli in the spike domain, *i.e.*, the time of spikes. Recent studies suggest that the responses of neurons in early visual systems to natural stimuli are highly precise [17]. We therefore push the previous question a step forward and ask:

- *How can visual information be represented and processed by time of spikes generated by massively parallel neural circuits with biophysical spiking neurons?*

Spikes are the results of proper stimulation applied to highly nonlinear dynamical systems. The dynamics of neurons and their biophysical spiking mechanisms have been well established due to the groundbreaking study by Hodgkin and Huxley [76]. Their work has influenced many computational investigations of large-scale biological neural networks, spike synchronization, and other systems. In studies of neural coding, however, many of the well-established biophysical spiking mechanisms were either ignored or largely replaced by simplified spiking models that cannot fully capture the dynamics and spike timing of their biophysical counterparts.

In this dissertation, we address the above shortcoming by investigating visual encod-

ing neural circuits in which spikes are generated by biophysically-grounded spiking neuron models. We argue that visual stimuli can be represented by the spike times governed by biophysical spiking mechanisms. Furthermore, we demonstrate a method to perform certain visual transformations on visual stimuli, *e.g.*, translations and rotations, directly in the spike domain.

Neural systems are intrinsically noisy. It is well known that spiking responses of biological neurons to the same stimulus vary from trial to trial. This variability mainly originates from intrinsic noise sources in, for example, neurons and synapses. The following question naturally arises in the study of stimulus representation by spiking neural circuits:

- *What is the effect of intrinsic noise sources on stimulus representation and on the precision of functional identification?*

In order to reduce the effect of the variability in spiking response, repeated trials of the same stimulus are often employed in experimental recordings to obtain a Peri-Stimulus Time Histogram (PSTH). The sensory systems, however, do not have the luxury to expose themselves to repeated stimuli in real life. On a single exposure to a stimulus, the effect of the spike variability caused by noise then becomes highly relevant to how well the stimulus can be represented. We formally characterize in this dissertation the effect of intrinsic noise sources on spike variability in neural circuits with biophysical spiking neurons. This leads to a systematic investigation of the effect of intrinsic noise sources on stimulus representation as well as on the precision of functional identification.

The previous questions concern the representation and processing of visual stimuli by massively parallel neural circuits. For a biological system, it is also interesting to functionally identify the massively parallel neural circuits. Simply providing a method for identification is not sufficient for such a large-scale neural circuit, but an evaluation of the quality of the parameter estimation is necessary. Unfortunately, the ground truth of biological systems is not available for the evaluation of identification quality.

It is necessary to ask:

- *What is an effective criterion to evaluate the quality of functional identification of a massively parallel spiking neural circuit?*

It is important to note that this question is different from the one that inquires about the evaluation of the identification of a single neuron. The difference and difficulty lies in that 1) the number of neurons is large and 2) the role of an entire massively parallel neural circuit is different from that of a single neuron, despite the fact that many single neurons constitute an entire circuit.

The above, however, hints at a possible solution that allows one to connect the problem of functional identification and its evaluation with the problem of encoding and decoding. We detail such an approach in this dissertation to address the above problem.

1.2 Approach

Marr proposed three levels of analysis in studying an information processing system [129], namely,

1. **computational level**: what does the system do and why?
2. **algorithmic/representational level**: representation that the system uses and processes it employs to build and manipulate the representation,
3. **physical level**: implementation of algorithm by biological/neural components.

This dissertation mainly focuses on the second level of analysis, *i.e.*, on the algorithmic level. By doing so, we highlight the functional role of the neural circuit.

We study the representation and processing of natural visual stimuli by massively parallel neural circuits under the formal framework of Video Time Encoding Machines (Video TEMs) and Video Time Decoding Machines (Video TDMs) [98]. Video TEMs are asynchronous signal encoders that encode analog visual scenes (movies, animation, etc.) in the time domain [98]. The amplitude domain information of analog stimuli is transformed into the time sequences of action potentials.

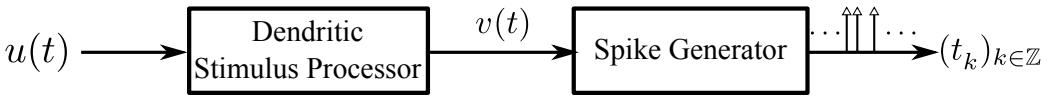


Figure 1.2: Model of single neural encoding in the framework of TEMs. A signal $u(t)$ is encoded by a neuron into a spike train $(t_k)_{k\in\mathbb{Z}}$, where k is the spike index. The neuron consists of a dendritic stimulus processor cascaded with a spike generator.

The study of functional identification of massively parallel neural circuits is based on Channel Identification Machines (CIMs) [102]. CIMs are algorithms that identify neural circuit parameters (e.g., receptive fields) directly from spike times generated in response to a collection of test stimuli.

In both frameworks, the response of an individual neuron is modeled as schematically shown in Figure 1.2. Here, a stimulus $u(t)$ is processed by a dendritic stimulus processor (DSP) cascaded with a spike generator (SG). The DSP (an example being a linear receptive field) models the processing taking place in the dendritic trees of the neuron and the network of neurons preceeding the neuron in stimulus processing. The spike generator models the axon hillock of the neuron in generating spikes.

Therefore, we largely ignore the physical-level implementations, for example, how synapses, dendrites, and a network of neurons implement a linear receptive field of a neuron. These are, however, important and interesting topics that are subjects of research that focuses on the implementation level [194]. By using biophysically realistic neuron models, we effectively consider the encoding with a detailed, circuit-level implementation of the spike generation mechanism. The analysis of the encoding by these neuron models, however, is carried out on the algorithmic level.

We note that the methodology presented in this dissertation (i) does not group the spikes into time bins to create a rate response, (ii) does not use the PSTHs and (iii) does not employ at any stage the theory of point processes. The methodology here, thereby, differs from the Linear-Nonlinear-Poisson (LNP) model (and its follow-ups) that is frequently applied in functional identification [146]. Rather, the modeling approach to neural encoding relies on methods of functional analysis and the theory of dynamical systems.

A key component throughout the study in this dissertation is the modeling of sensory stimuli in Reproducing Kernel Hilbert Spaces (RKHSs). This enables the study of visual representation by spikes in a similar way as in the study of sampling theorems, and provides a geometric interpretation to the neural encoding problem. A spectrum of techniques developed in the study of RKHSs can then be applied to the reconstruction of visual stimuli and the functional identification of circuits.

Another mathematical theory on which we rely heavily is the Volterra series expansion [165]. A Volterra series is a function expansion of a dynamic, nonlinear, time-invariant functional. Formally, for a continuous time-invariant system with input $u(t)$, its output $v(t)$ can be expressed as a Volterra series as

$$v(t) = h_0 + \sum_{p=1}^{P} \int h_p(s_1, \cdots, s_n) u(t - s_1) \cdots u(t - s_p) ds_1 \cdots ds_p, \tag{1.1}$$

where $h_p(t_1, \cdots, t_p), p = 1, 2, \cdots, P$, are the pth order Volterra kernels.

The Volterra approach provides a method to study different forms of representation of visual stimuli in early visual systems. For example, we will study, in addition to the representation of monochromatic visual stimuli, how color visual stimuli can be encoded and decoded. Furthermore, it is possible to investigate the representation by neurons with nonlinear processing capability in the early visual system, *e.g.*, complex cells in V1.

To study the role of the biophysical spiking mechanism in representations of visual information, we employ the theory of dynamical systems. We start with ordinary differential equations that describe the dynamics of biophysical spike generators. By studying the dynamical system, it is possible to incorporate intrinsic noise sources into the investigation by extending the system to stochastic differential equations. Studies of the effects on spike variability and on precision of stimulus representation and functional identification becomes possible. This differs from the approach that assumes that the system is probabilistic in nature.

The work presented here requires a substantial amount of investment in the mathematical formalism employed throughout. Functional analysis is typically employed in understanding the nature of encoding. Utilization of techniques such as optimization and regularization is a recurring scheme in formulating the decoding and functional identification problems.

The theoretical analyses in this thesis are accompanied with computational realizations. The computational requirements of the massively parallel neural circuits are met by high performance parallel computing based on Graphics Processing Units (GPUs). The utilization of GPU clusters is an essential component throughout this dissertation.

We also note that the visual system is a part of a highly interconnected network of systems in the brain. It is regulated, modulated, and controlled by higher brain areas and motor systems. We ignore, for example, the role of attention in the representation, while these are valid and important subjects. By studying the early visual systems, the effect of these sources of feedbacks on the direct visual representation is minimized. As in control theory, it is a standard practice to study first the open-loop system and understand its function before introducing the feedbacks.

Similar to the approach taken in [177], a key recurring scheme is that we will show the videos reconstructed from neural activities rather than showing how much information is contained in the neural response, a measure that typically appears in information theoretical analysis. While important in characterizing the coding capability of neural responses, an information theoretical measure does not indicate directly that a painting that the visual system sees is a portrait of the *Mona Lisa*, for example. By showing the videos, we effectively build an "oscilloscope" for the massively parallel neural circuits such that the information can be probed and compared to the original. This provides a direct, intuitive method to assess how much information is carried by the neural activities.

1.3 Contributions and Organization of the Dissertation

This dissertation mainly contributes to extend the framework of Video TEMs/TDMs and that of CIMs in the following ways.

1. First, we demonstrate that visual stimuli encoding neural circuits in which spikes are generated by biophysically-grounded spiking neuron models can be characterized as generalized sampling by introducing the PRC manifold for BSGs. Conditions of perfect recovery of visual stimuli from spike times are provided.

2. Second, we show that the Video TEMs are scalable to massively parallel neural circuits

and show that Video TDMs under massively parallel settings are computationally tractable.

3. Third, we develop models accounting for neural encoding in color and stereoscopic vision and derive a decoding algorithm demonstrating that under certain conditions, color/binocular information that is mixed into an unlabeled set of spike trains can be reconstructed/demixed.

4. Fourth, we provide functional identification algorithms to estimate circuit parameters of massively parallel neural circuits encoding color/stereo visual stimuli, and provide evaluation criteria for the functional identification of the entire circuit by incorporating intuitive comparisons in the stimulus space via decoding.

5. Fifth, we extend neural circuits to account for nonlinear stimulus processing in addition to the nonlinear biophysical spiking mechanism and show that intrinsic noise sources arising in the nonlinear circuits can be rigorously and systematically investigated.

6. Sixth, we extend the Video TEM to model nonlinear processing occuring in early vision, *e.g.*, in the complex cells, we provide an efficient decoding algorithm demonstrating that an ensemble of complex cells can faithfully represent visual stimuli, and we extend the CIMs to identify complex cells' nonlinear energy processing in a computationally tractable way.

7. Finally, we demonstrate that identity preserving transformations of visual stimuli can be performed in the spike domain by switching mechanisms that regulate the connection between Video TEMs and Video TDMs.

Together, this thesis presents a unified theoretical framework to comprehensively study massively parallel spiking neural circuits arising in early vision. We briefly outline this dissertation and the contributions of each chapter in the following.

We begin by introducing Video TEMs and CIMs in Chapter 2. We briefly review a number of models of neural encoding, of methods in neural decoding and of methods in functional identification.

In Chapter 3, we introduce and investigate massively parallel neural circuit architectures encoding natural visual scenes with neuron models consisting of dendritic stimulus processors in cascade with biophysical spike generators. On the encoding side, we characterize the BSG I/O with a phase response curve (PRC) manifold and interpret neural encoding as generalized sampling in an RKHS. We provide a decoding algorithm that recovers visual stimuli encoded by a massively parallel neural circuit. We extend the architecture to encompass neuron models with ON-OFF biophysical spike generators with self- and cross-feedback. Algorithms based on recurrent neural networks are proposed to solve the decoding problem in a more distributed fashion. Combined with a stitching algorithm, the decoding of visual stimuli encoded by massively parallel neural circuits is demonstrated. We investigate the performance of signal recovery in the Nyquist region and for different temporal bandwidths.

Chapter 4 and Chapter 5 address the massively parallel neural circuits in color and stereoscopic vision. We present in Chapter 4 Color Video Time Encoding Machines (Color Video TEMs) for encoding color visual stimuli that take into account a variety of color representations within a single neural circuit. We then derive a Color Video Time Decoding Machine (Color Video TDM) algorithm for color demixing and reconstruction of color visual scenes from spikes produced by a population of visual neurons. Extension to stereoscopic color vision is provided.

In Chapter 5, we formulate Color Video Channel Identification Machines (Color Video CIMs) for functionally identifying color visual processing performed by a spiking neural circuit. Furthermore, we derive a duality between TDMs and CIMs that unifies the two and that leads to a general theory of neural information representation for stereoscopic color vision. We provide examples demonstrating that a massively parallel color visual neural circuit can first be identified with arbitrary precision, and its spike trains can be subsequently used to reconstruct the encoded stimuli. We argue that evaluation of the functional identification methodology can be effectively and intuitively performed in the stimulus space by decoding using identified circuit parameters.

Chapter 6 extends the encoding neural circuit in two ways. First, the DSPs are extended to handle nonlinear stimulus processing and nonlinear spike processing. Second, we investigate two intrinsic noise sources arising (i) in the active dendritic trees underlying the

DSPs, and (ii) in the ion channels of the BSGs. We show that encoding with a neuron model consisting of a nonlinear DSP cascaded with a BSG with intrinsic noise sources can be treated as generalized sampling with noisy measurements. We theoretically analyze the effect of noise sources on stimulus decoding and on the precision of functional identification. We demonstrate through extensive simulations the effects of noise on encoding stimuli with circuits that include neuron models that are akin to those commonly seen in sensory systems, e.g., complex cells in V1. For simplicity, the characterization was investigated for a parallel neural circuit encoding temporal stimuli. We provide at the end examples in visual stimuli encoding as in Chapter 3, but with intrinsic noise sources.

In Chapter 7, we focus on a key nonlinear model of visual stimulus encoding in V1 that has been put forth in Chapter 6. We investigate the capability of a population of complex cells to encode space-time-varying stimuli and demonstrate how encoded stimuli can be recovered from spike times with much-reduced number of spikes/neurons than using the formalism in Chapter 6. The reconstruction algorithms are formulated as rank minimization problems. This shows that visual representation with complex cells is achievable using biologically realistic neural resources. We also show how to identify energy processing in complex cells and demonstrate that our identification algorithms substantially outperform STC. Finally, we evaluate our identification algorithms by reconstructing novel stimuli in the input space.

After studying the representational capability of massively parallel neural circuits in the spike domain, we investigate neural circuit architectures for identity preserving transformations (IPTs) on visual stimuli in the spike domain in Chapter 8. A number of IPTs are demonstrated including faithful stimulus recovery, as well as simple transformations on the original visual stimulus such as translations, rotations, and zooming. Using group theoretic methods, we advance two different neural encoding architectures and demonstrate the realization of exact and approximate IPTs. These are realized in the spike domain processing block by a “switching matrix" that regulates the input/output connectivity between the stimulus encoding and decoding blocks [151].

This dissertation is concluded in Chapter 9. We describe possible future research directions.

Chapter 2

Models of Neural Representation of Visual Stimuli in Early Visual Systems

In this chapter, we briefly review models of neural representation of visual stimuli in the early visual systems. We start with introducing the Video Time Encoding Machines and the Channel Identification Machines. We then review existing models in neuroscience of neural encoding / decoding as well as methods of functional identification.

2.1 Video Time Encoding Machines

Video Time Encoding Machines (Video TEMs) are asynchronous visual stimuli encoders that encode analog visual scenes into multidimensional spike trains [98]. A schematic diagram of a Video TEM investigated previously is shown in Figure 2.1(a). The Video TEM consists of M neurons encoding a visual stimulus $u(x, y, t)$ into spike trains $(t_k^i)_{k\in\mathbb{Z}}, i = 1, 2, \cdots, M$. Each neuron is modeled as a linear receptive field $h^i(x, y, t)$ cascaded with a spike generator. Previously, spike generators modeled by simplified spiking neuron models have been investigated, such as Leaky-Integrate-and-Fire (LIF) and Threshold-and-Fire (TAF) neurons with feedback [98, 112].

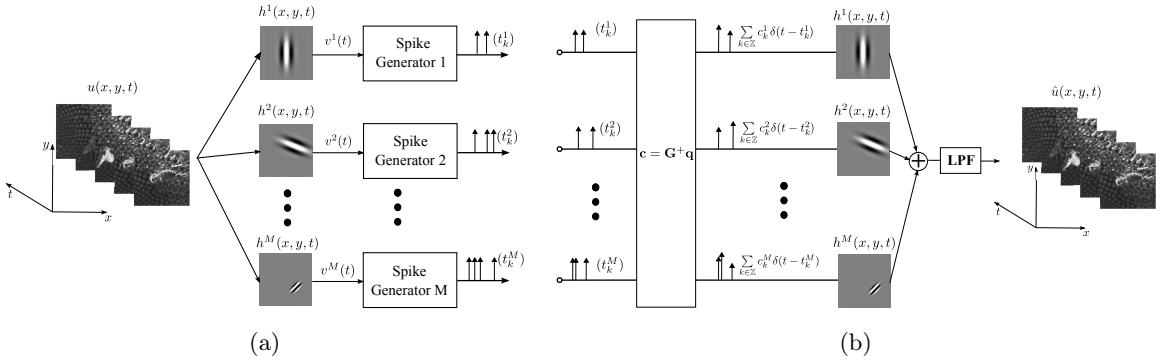


Figure 2.1: Schematic diagram of a Video TEM and a Video TDM.

For Video TEMs built with linear receptive fields cascaded with IAF neurons, the encoding of the visual stimuli can be formally interpreted as generalized sampling by expressing the encoding as a set of inner product operations in a Hilbert space [98]

$$\langle u, \phi_k^i \rangle = q_k^i, k \in \mathbb{Z}, i = 1, \cdots, M, \tag{2.1}$$

where u is an element of the space of temporally bandlimited functions in $\mathbb{L}^2(\mathbb{R}^3)$ (the square integrable functions on $\mathbb{R}^3$), $\phi_k^i, k \in \mathbb{Z}, i = 1, \cdots, M$ are sampling functions derived from the operations of the linear receptive field and IAF neuron, and $q_k^i, k \in \mathbb{Z}, i = 1, \cdots, M$ are the corresponding measurements of the sampling functions. Both the sampling functions and the measurements can be determined by the spike times $(t_k^i)_{k\in\mathbb{Z}}, i = 1, \cdots, M$.

Video Time Decoding Machines (Video TDMs) reconstruct visual scenes from spike times generated by Video TEMs. A schematic diagram of a Video TDM is shown in Figure 2.1(b). To reconstruct visual stimuli encoded by the Video TEM with linear receptive fields cascaded with IAF neurons, u is assumed to have the form

$$u(x,y,t) = \sum_{i=1}^{M} \sum_{k\in\mathbb{Z}} c_k^i \psi_k^i(x,y,t), \tag{2.2}$$

where $\psi_k^i(x,y,t) = (h^i(x,y,\cdot) * \mathrm{sinc}(\cdot - t_k^i))(t)$, with $\mathrm{sinc}(t) = \sin(\Omega_t t)/(\pi t)$ and Ω_t is the bandwidth, and $*$ denotes the convolution operation. The coefficients $c_k^i, k \in \mathbb{Z}, i = 1, 2, \cdots, M$ can be obtained by plugging (2.2) into (2.1) and solving the system of linear equations.

If the spatial components of the receptive fields are picked from a spatial wavelet filter bank and the temporal components have full frequency support, then the Video TDM can perfectly reconstruct the visual stimuli $u(x, y, t)$ provided that the spike density and the number of neurons are sufficiently large [98].

2.2 Channel Identification Machines

Channel Identification Machines (CIMs) are algorithms that identify neural circuit parameters (e.g., receptive fields and parameters of spike generators) directly from spike times generated in response to a collection of test stimuli [102].

CIMs are formal dual of Time Decoding Machines [102, 103]. In the case of the Video TEM described in Section 2.1, a CIM can be utilized to identify a linear receptive field $h(x, y, t)$ (omitting i) by presenting to the encoding circuit M different visual stimuli $u^i(x, y, t), i = 1, \cdots, M$. The CIM states that the encoding of a stimulus u^i by a neuron with linear receptive field h cascaded with a spike generator can be equivalently viewed as encoding of a "stimulus" $\mathcal{P}h$ by a neuron with a "linear receptive field" u^i cascaded with the same spike generator. Here, $\mathcal{P}h$ denotes the projection of h onto the space of input stimuli. This equivalence is evident by comparing the diagram shown in Figure 2.2(a) and Figure 2.1(a).

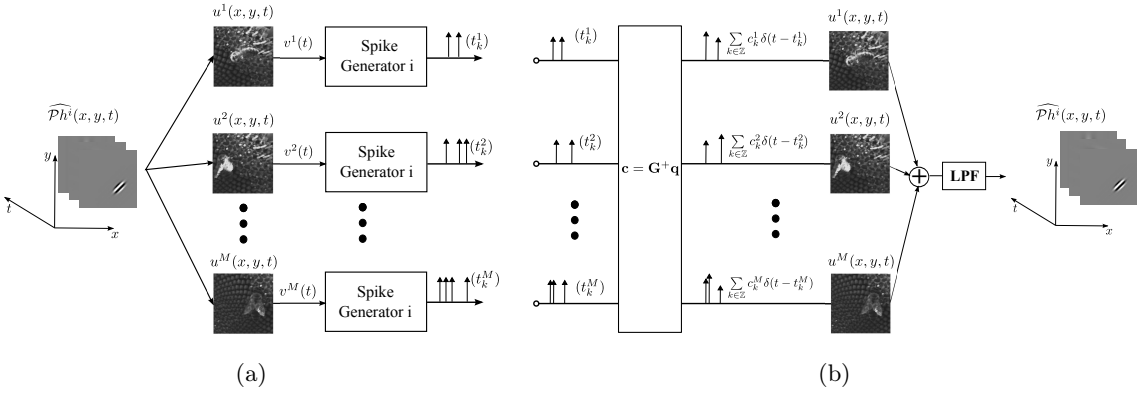


Figure 2.2: Duality between CIM and TDM. (a) Alternative interpretation of encoding in Channel Identification Machines. (b) Block diagram of CIM algorithm.

Functional identification of the projection of h then becomes a decoding problem where

the signal to be decoded is the projection of h. As a result, the identification algorithm in the CIM can be formulated similarly to the decoding algorithm in the Video TDM, as shown in Figure 2.2(b). An important result stated by the CIM is that only the projection of the linear receptive field in the stimulus space can be properly identified [102]. This suggests that function identification is dependent on the stimuli used to probe the system.

Some of the key advantages of parameter identification in the setting of CIMs are (i) clear lower bounds on the number of test stimuli and spikes required for identification can be specified and (ii) both synthetic and naturalistic stimuli can be used [103, 104].

2.3 Existing Methods in Neural Encoding, Decoding and Functional Identification

2.3.1 Encoding Models

2.3.1.1 Linear-Nonlinear-Poisson Model

A widely used encoding model in the visual neuroscience is the Linear-Nonlinear-Poisson (LNP) model [26, 145]. The LNP model consists of 3 cascaded blocks as shown in Figure 2.3.

Figure 2.3: Block diagram of the LNP model.

In the LNP model, stimulus u is first passed through a linear filter followed by a static nonlinearity that maps the filtered output into a spiking rate. Subsequently, spikes are generated by a Poisson spike generator where spiking is a result of an inhomogeneous Poisson process with a variable rate set according to the spiking rate. Such spiking mechanism, however, differs from actual biophysical spiking mechanism taking place in the membrane of biological neurons.

The LNP models can be extended to have multiple linear filters [169]. With a squaring nonlinearity, the multi-dimensional LNP model were used to model stimulus processing in

the complex cells in V1 [22].

2.3.1.2 Generalized Linear Model

The Generalized Linear Model (GLM) is a generalization of the LNP model [145, 149]. Components introduced in the GLM model in addition to the LNP models are post-spike filters and coupling filters. The post-spike filters are self-feedback filters that take spikes generated by the Poisson spike generator as inputs. The outputs of them are added to the linearly filtered stimuli before going through the static nonlinearity. The coupling filters are cross-feedback filters that influence other neurons instead of the neuron that generates the spikes. Other variations include replacing the Poisson spike generator with an IAF or LIF neuron [146, 148].

The main purpose of introducing the post-spike filters is to create an refractory effect to mitigate inability of the Poisson process in doing so. The cross-feedback filters enables the study of coupling between two neurons.

2.3.1.3 Virtual Retina Model

Unlike the Video TEM, LNP or GLM models, the Virtual Retina models the retinal encoding with a more detailed, layered structure resembling the network structure in the retina [198]. It includes a layer of horizontal cells, a layer of bipolar cells and a layer of retinal ganglion cells. A key feature of the Virtual Retina Model is that the dynamics of the bipolar cell layer are governed by differential equations with interaction among neighbors both in space and in time. The nonlinearity built into the bipolar cell layer allows the model to exhibit contrast gain control as observed in experiments [7, 45, 171].

The Virtual Retina model provides a platform to generate realistic spiking output from the retina. As the modeling is on the implementation level, the analytic tractability of the retina I/O is currently less clear. As the study in this dissertation focus on the algorithmic level, it allows to characterize the I/O of visual systems in an analytically tractable way.

2.3.1.4 Neuromorphic Encoding Models

Neuromorphic approaches for building silicon retinas have also been proposed [154]. These approaches also aim to study biological visual encoding on the implementation level. The use of event driven silicon components allows for high dynamic range in imaging natural visual environments whose light intensity can vary several orders of magnitude in a short spatial range. The model, however, does not currently take into account the extensive processing in the retina as exemplified by receptive fields. More carefully fitted neuromorphic imaging devices, such as the neuromorphic retina [203], and the artificial retina [125] built upon the Dynamic Vision Sensor (DVS) [121], aim to reproduce many spatial and temporal properties of RGC responses [33].

As we have mentioned in the introduction, many of the underlying encoding schemes in neuromorphic approaches are still empirical. As a result, it is less clear how the spikes generated by those neuromorphic systems serve as a representation of the stimulus analytically.

2.3.2 Decoding Methods

2.3.2.1 Optimal Linear Decoder

The optimal linear decoder applies to decoding from spiking rate of an ensemble of neurons [192]. In the case of temporal stimulus, the reconstruction of a stimulus $u(t)$ from response of M neurons, $r^i(t), i = 1, 2, \cdots, M$, is of the form

$$u(t) = a + \sum_{i=1}^{M} \int r^i(s) h^i(t-s) ds,$$

where $h^i(t), i = 1, 2, \cdots, M$ are optimal linear reconstruction filters and a is a constant. The optimal linear reconstruction filters are constructed in controlled experiments where the zero-mean stimulus $u(t)$ and mean subtracted neural response $r(t)$ are observable, and by minimizing the mean square error between the stimulus and its estimate. The optimal linear reconstruction filters amounts to

$$h^i(t) = \mathcal{F}^{-1}\left[\frac{S_{RU}(\omega)}{S_{RR}(\omega)}\right],$$

where $\mathcal{F}^{-1}$ is the inverse Fourier transform, S_{RR} is the power spectrum of the neural response and S_{RU} is the cross spectrum between the stimulus and the response.

Note, however, that neural encoding is highly nonlinear due to the spike generation mechanism. It becomes difficult to use a linear decoder to achieve good decoding performance without taking into account the nonlinear behavior of the neurons. Furthermore, it is questionable that a set of reconstruction filters can provide decoding for any stimulus as the stimulus space is typically undefined or can be extraordinarily large.

2.3.2.2 Bayesian Decoding

Bayesian approaches in decoding neural responses typically takes the form of maximum a posteriori (MAP) estimate of the stimulus [145, 150]. The objective is to maximize the posterior over stimulus u given the neural response r, *i.e.*, to maximize

$$p(u|r) = p(r|u)p(u).$$

This is equivalent to obtain

$$\hat{u} = \underset{u}{\operatorname{argmax}} \left\{ \log p(u|r) + \log p(u) \right\}.$$

Note that in order to make the above problem tractable, the stimulus prior needs to be log-concave. The method, however, have not been applied to biophysical spiking neuron models.

2.3.2.3 Rank Order Coding

Rank order coding was proposed based on the temporal constraint of neural coding [186]. Human vision can perform complex tasks such as recognition in about 150 ms. Any important feature of the visual scene has to be encoded in the retina and sent to the cortex in much less than that amount of time, since the propagation of spikes and more demanding computation for recognition also take time. Such a concept has been supported by experiments in the primate retina [69].

Under such constraint, rank order coding proposed that the spikes that are generated earlier in the retina after an image onset contain more important information about the image. Therefore, image can be reconstructed progressively in real time as more neurons generate spikes, and the spikes that contains more information are used to decode first. The

complexity of decoding can thus be kept low while the recovery quality and the speed of reconstruction is high.

A look-up table (LUT) can be computed by analyzing the rank and activation levels of the neurons in response to a large collection of natural images. The LUT can then be used to approximate the activation values to reconstruct stimulus by assigning the approximate activation values to neurons with higher rank.

A drawback of such a decoding is that the decoding scheme implies a global reset mechanism to create a reference point when the activation of neuron, or time of spikes, can be compared to. Therefore, it can be applied to decode discrete images only. Vision, however, is continuous. We will demonstrate the decoding of continuous visual stream in this dissertation.

2.3.3 Methods of Functional Identification

Most of the algorithms used in computational neuroscience deals with functional identification. We briefly describe two most widely used techniques in identification using the LNP model.

2.3.3.1 Spike Triggered Average

The Spike Triggered Average (STA) is among the most commonly employed methods in functional identification. It is typically used to estimate the linear receptive field in the LNP model [36]. Given the spike times (t_k) encoding a zero mean stimulus u, the STA provides an estimate $\hat{h}$ of the linear receptive field by

$$\hat{h}(t) = \frac{1}{N}\sum_{k=1}^{N} u(t_k - t),$$

where N is the number of spikes. Note that a lot of spikes are required to produce a good estimate for the true expectation.

The STA is closely related to reverse correlation methods. However, white noises stimuli are usually required in the STA to provide an unbiased estimate of the filter for the LNP model with a single linear filter [169]. The use of natural stimuli cannot guarantee a consistent identification of the filter.

2.3.3.2 Spike Triggered Covariance

For the LNP models with multiple linear filters such as models for complex cells, the STA will not provide a proper estimate of the filters. The filters may, however, be recovered from how the variance of the spike-triggered ensemble differs from that of raw ensemble [169]. This pertains to estimating the covariance by

$$c_{STC}(t,s) = \frac{1}{N-1}\sum_{k=1}^{N}\left[u(t_k - t) - \hat{h}(t)\right]\left[u(t_k - s) - \hat{h}(s)\right],$$

where N is the number of spikes and $\hat{h}(t)$ is the STA.

In practice, the covariance is computed in the discrete space and represented by a finite dimensional matrix. The eigenvectors of the STC matrix associated with the eigenvalues with the largest absolute values provide an identification of multiple linear filters. The covariance method is closely related to the Volterra approach that will be detailed in this dissertation. Note that the STC suffers from the same problem as in the STA. They require a large number of spikes to achieve a good identification.

Chapter 3

Reconstruction of Natural Visual Scenes from Spike Times

As we have mentioned earlier, many computational approaches rely on simplified spiking models that poorly approximate the dynamics of well established biophysical spiking mechanism. In this chapter, we address this problem and investigate visual encoding neural circuits where spikes are generated by biophysically-grounded spiking neuron models. Our work is motivated in part by an increasing interest in the coding and computational capability of biophysical spike generators [3, 49, 127, 167]. The difficulty of studying the representational and computational ability of conductance based spike generators, however, lies in their highly nonlinear nature [81, 134].

The spatio-temporal neural encoding circuits studied here for encoding natural visual scenes comprise neuron models consisting of dendritic stimulus processors (DSPs) in cascade with biophysical spike generators (BSGs). The highly nonlinear behavior of BSGs require novel methods of analysis of neural encoding circuits and novel decoding algorithms for signal recovery. Previous decoding methods relied on the weak signal assumption: the trajectory of the state of the BSG is geometrically in the neighborhood of a single limit cycle and functionally a phase response curve (PRC). This assumption no longer holds for visual scenes. On the encoding side we functionally characterize the I/O of the BSG with a

This chapter appears, in part, in [110] ©2014 IEEE.

PRC manifold and provide insights on the role of BSGs in visual encoding. Encoding can then readily be interpreted as generalized sampling.

For decoding we derive an algorithm that recovers the visual stimuli encoded by a neural circuit. We give conditions on perfect reconstruction of natural visual scenes. This greatly simplifies the characterization of the complexity of the neural circuit needed to faithfully encode visual stimuli.

We also put forth a biologically relevant neural circuit architecture that consists of ON and OFF component BSGs, that are inspired by the ON and OFF cells present in early visual systems. With the help of a PRC manifold, we show that decoding is tractable even for a wide signal dynamic range. As a result, bias currents that were essential in the encoding process previously formulated in the literature can largely be reduced or eliminated.

Finally, we provide extensive simulations of massively parallel encoding and decoding of visual scenes on a cluster of GPUs. We evaluate investigate the performance of signal recovery in the Nyquist region and for different temporal bandwidths.

In Section 3.1, we present the neural circuit architecture for encoding visual scenes, model the space of natural scenes and formally define the operation of its component DSPs and BSGs. In Section 3.2, we derive the I/O relationship of the encoding neural circuit using the PRC manifold. In Section 3.3, we provide a decoding algorithm and conditions on perfect reconstruction. We also provide extensions to encoding circuits that incorporate ON-OFF neuron pairs with feedback. In Section 3.4, we present algorithmic techniques to perform decoding of visual stimuli with large screen size on a GPU cluster. Finally, in Appendix 3.5, we will present numerous examples of decoding visual scenes backed-up by video recovery performance demonstrations.

3.1 Modeling Neural Encoding of Natural Scenes

In this section we present a class of neural circuit architectures for encoding visual scenes (signals) into spike trains. We define the input space, describe each component in the neural circuit and formally define their operation and dynamics.

3.1.1 Architecture for Neural Encoding of Natural Scenes

Unlike digital cameras that sample at a single point (pixel), the projection of light intensity onto a two-dimensional plane, individual neurons in the visual system respond to a region of the visual space. In particular, each neuron may respond maximally (in terms of firing rate) to certain spatio-temporal patterns of illumination of the visual stimuli. Such spatio-temporal illumination patterns are then the most effective stimuli that drive a neuron to fire action potentials [36] and are referred to as *receptive fields*. Receptive fields reflect the processing performed on visual scenes by the visual system and studies of visual neurons are typically accompanied by their characterization [79, 93]. Many theories have been proposed to explain the processing capability of certain types of receptive fields in early visual systems, *e.g.*, decorrelation of outputs, efficient information transmission and sparsity [72, 140].

We begin by introducing the encoding of visual scenes by a single *spiking* neuron. The encoding is modeled by a spatio-temporal dendritic stimulus processor (DSP) cascaded with a biophysical spike generator (BSG). As schematically illustrated in Figure 3.1, the neuron encodes a visual signal (scene) $u(x, y, t)$ into a train of spikes whose time occurrences are denoted as $(t_k^i)_{k\in\mathbb{Z}}$, where i denotes the index of neurons and k denotes the index of spikes.

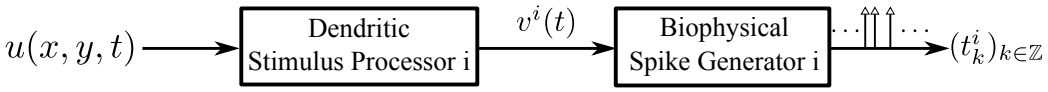


Figure 3.1: Diagram of a single model neuron consisting of a spatio-temporal dendritic stimulus processor in cascade with a biophysical spike generator. The neuron encodes a visual scene $u(x, y, t)$ into a train of spikes whose time occurrences are $(t_k^i)_{k\in\mathbb{Z}}$. (From [110] with permission, ©2014 IEEE).

The spatio-temporal DSP, as a functional model of processing underlying the receptive fields, describes the aggregated effects of stimuli on active dendritic integration. It is important to note that the actual processing of stimuli does not only take place in the dendritic trees, but it is also the result of actions by a complex network of neurons proceeding the neuron of interest. However, the aggregated effect emerges on the dendrites of the neuron in question. Therefore, the DSPs serve as functional models of *processing* on *stimuli*

up to and including the *dendritic* tree of the neuron. The importance of DSPs as part of the neuron model expresses the belief that with proper network connectivity the major portion of processing is performed at neuron's dendritic level [124, 174]. However, as a functional model, the DSPs considered here do not address the implementation of processing by neural/dendritic tree components. In this chapter, the DSP considered is a linear spatio-temporal receptive field [36].

The BSG models spike generation at the axon hillock level, where neurons respond, in the form of action potentials, to synaptic currents aggregated by active dendritic trees [60]. Many neuromorphic devices and visual system models were based on simplified, phenomenological models such as LIF and TAF neurons [98, 121, 198]. The biophysical mechanisms of spike generation have been well studied (see [81] for a review) and paved the way for investigating the representation of visual scenes using spike times.

A single neuron is not sufficient to represent an entire visual scene. An ensemble of neurons with different receptive fields tiling the visual space is needed. Figure 3.2 schematically illustrates a class of neural circuit architectures for encoding visual scenes (signals). A space-time varying signal $u(x, y, t)$ is encoded by a parallel neural circuit consisting of M neurons each composed of a spatio-temporal DSP cascaded with a BSG. The output of the neural circuit is a multi-dimensional spike train whose spike times are observable.

The neural encoding model of visual scenes in Figure 3.2 arises naturally in studying the early visual system of vertebrates including the retina, the lateral geniculate nucleus (LGN) and the primary visual cortex (V1). We use the primate retina here as an example. Retinal ganglion cells (RGCs), the outputs of the retina, generate action potentials (spikes) and transmit them to higher brain areas [130]. Although recent studies suggest that some bipolar cells may generate action potentials [8, 168], most neurons preceding the RGCs process the visual input through analog, graded membrane potentials [178]. As these intermediate neurons synapse onto the dendrites of RGCs, the aggregated effect of (analog) processing in the intermediate neurons and in the dendritic trees of the RGCs is modeled as spatio-temporal DSPs. The axon hillock of the RGCs is modeled as a spike generator. For LGN and V1, the spatio-temporal DSPs model the aggregated effects of the overall preprocessing of the visual stimuli, in analog and in spike domain, preceding the spike generation by neurons

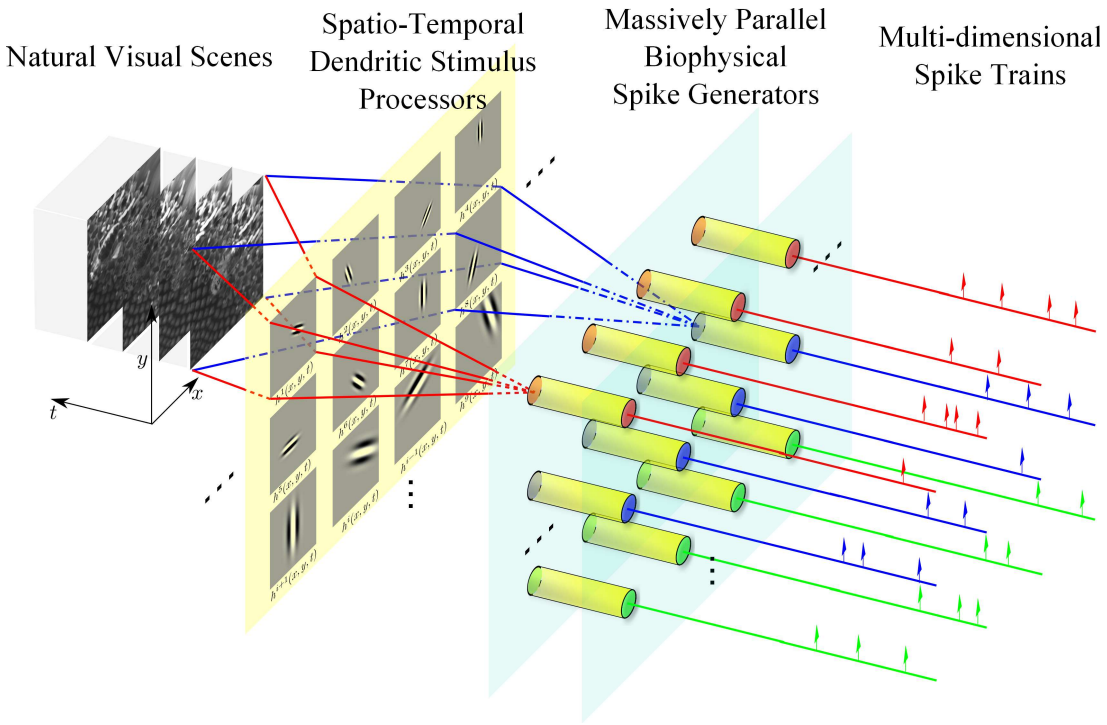


Figure 3.2: Schematic diagram of a neural circuit architecture for encoding natural scenes into multi-dimensional spike trains. The circuit consists of two cascaded stages: spatio-temporal dendritic stimulus processors, and biophysical spike generators (see text in Section 3.1.1 for more details). (From [110] with permission, ©2014 IEEE)

in the corresponding visual areas.

The neural circuit depicted in Figure 3.2 encoding visual scenes into spike times can be seen as an instantiation of a Video TEM [98]. In sections 3.1.3 and 3.1.4, we describe the main components of the Video TEM comprising neuron models with linear DSPs [98] and biophysical spike generators. The use of conductance based, biophysical spike generators extends several analytically tractable spiking models including Integrate-and-Fire (IAF) neurons, Threshold-and-Fire (TAF) neurons and their derivatives [98, 112]. The biophysical spike generators model the dynamics of spike generation more precisely and therefore, lead to a better understanding of encoding by biological neural circuits. But first we model in Appendix 3.1.2 natural visual scenes with a view towards computational tractability.

3.1.2 Modeling the Space of Natural Visual Scenes

We first describe a model input space for natural visual scenes. The input space model is the bedrock for formal sampling theorems [170]. For example, the space of bandlimited signals poses a limit on the minimum uniform sampling rate (the Nyquist rate) for faithful signal representation. The modeling of the space of input signals is usually less emphasized in studying responses of single neurons in the visual system. However, it arises naturally when studying an entire neural circuit as well as single neurons in the context of encoding and decoding of time-varying signals and visual scenes [98, 106, 112].

In what follows, we denote the intensity of a space-time-varying visual signal (scene) on a domain $\mathbb{D}$ as $u = u(x, y, t)$. All visual signals are assumed to be elements of the vector space $\mathcal{H}_3^s$ of trigonometric polynomials.

Definition 1. *The space of trigonometric polynomials $\mathcal{H}_3^s$ is a function space whose elements are functions defined on the domain*

$$\mathbb{D} = \mathbb{D}_x \times \mathbb{D}_y \times \mathbb{D}_t = [0, S_x] \times [0, S_y] \times [0, S_t],$$

of the form

$$u(x, y, t) = \sum_{l_t=-L_t}^{L_t} \sum_{l_x=-L_x}^{L_x} \sum_{l_y=-L_y}^{L_y} u_{l_x l_y l_t} e_{l_x l_y l_t}(x, y, t), \tag{3.1}$$

where

$$e_{l_x l_y l_t} = \left[\frac{1}{S_x S_y S_t}\right]^{1/2} e^{jl_x \frac{\Omega_x}{L_x} x + jl_y \frac{\Omega_y}{L_y} y + jl_t \frac{\Omega_t}{L_t} t} \tag{3.2}$$

is a set of orthonormal basis functions. Here, by abuse of notation, $u_{l_x l_y l_t}$ are the coefficients, $\Omega_x, \Omega_y, \Omega_t$ are the bandwidth of the space, and L_x, L_y, L_t are the order of the space in each variable.

Remark 1. *The use of the subscript in the notation of the space $\mathcal{H}_3^s$ indicates that the domain of elements in $\mathcal{H}_3^s$ is 3 dimensional; the use of superscript s (stimulus) indicates that $\mathcal{H}_3^s$ is associated with the stimulus. In Section 6, we will introduce a Hilbert space associated with the spikes, and it will be denoted with superscript a (action potential), i.e., $\mathcal{H}^a$.*

The visual signals $u \in \mathcal{H}_3^s$ are periodic in each variable and the periods amount to, respectively,

$$S_x = \frac{2\pi L_x}{\Omega_x},\ S_y = \frac{2\pi L_y}{\Omega_y},\ S_t = \frac{2\pi L_t}{\Omega_t}.$$

Note that the actual visual signal can be embedded into a subset of the domain of u to avoid periodicity. In general, elements in $\mathcal{H}_3^s$ are complex-valued. We assume that the elements of the visual space are real-valued functions, *i.e.*, $u_{-l_x,-l_y,-l_t} = \overline{u_{l_x l_y l_t}}$, where $\overline{u}$ denotes the complex conjugate of u.

More generally, $\mathcal{H}_3^s$ is a reproducing kernel Hilbert space (RKHS) [12, 188] endowed with the inner product

$$\langle u, w \rangle = \int_{\mathbb{D}} u(x,y,t)\overline{w(x,y,t)}dxdydt. \tag{3.3}$$

The reproducing kernel of $\mathcal{H}_3^s$ is

$$\begin{aligned} &K(x',y',t';x,y,t) \\ &= \sum_{l_t=-L_t}^{L_t} \sum_{l_x=-L_x}^{L_x} \sum_{l_y=-L_y}^{L_y} e_{l_x l_y l_t}(x-x', y-y', t-t'), \end{aligned} \tag{3.4}$$

and it can be easily verified that the reproducing property

$$\langle u, K_{xyt} \rangle = u(x,y,t), \forall u \in \mathcal{H}_3^s \tag{3.5}$$

holds, where $K_{xyt}(x',y',t') = K(x',y',t';x,y,t)$. It suggests that the inner product of $u \in \mathcal{H}_3^s$ with the reproducing kernel is the evaluation of u at (x,y,t). The inner product above has another simple interpretation: a signal u filtered with a low pass filter with impulse response K_{xyt} remains unchanged.

The RKHS $\mathcal{H}_3^s$ provides a general easy to use framework for modeling neural encoding of natural scenes. $\mathcal{H}_3^s$ is a natural discretization of the space of bandlimited functions on $\mathbb{L}^2(\mathbb{R}^3)$, where $\mathbb{L}^2$ denotes the space of square integrable functions. By letting $L \to \infty$, the elements of $\mathcal{H}_3^s$ converge to bandlimited functions. $\mathcal{H}_3^s$ is computationally highly effective in modeling natural visual scenes [112]. Moreover, the coefficients $u_{l_x l_y l_t}$ are closely related to the Fourier transform of the signal $u(x,y,t), (x,y,t) \in \mathbb{D}$. In practice, they can be approximated by the Fourier components of u and the Fast Fourier Transforms (FFT) can be employed for efficient computation of these coefficients.

In what follows we denote the total dimension of $\mathcal{H}_3^s$ by $dim(\mathcal{H}_3^s) = (2L_x + 1)(2L_y + 1)(2L_t+1)$, the spatial dimension of $\mathcal{H}_3^s$ by $dim_{xy}(\mathcal{H}_3^s) = (2L_x+1)(2L_y+1)$ and the temporal dimension by $dim_t(\mathcal{H}_3^s) = 2L_t + 1$. Clearly, $dim(\mathcal{H}_3^s) = dim_{xy}(\mathcal{H}_3^s) \cdot dim_t(\mathcal{H}_3^s)$.

3.1.3 Spatio-Temporal DSPs

The receptive fields of many types of neurons in early visual systems can be characterized by linear filters [9, 36, 136]. For example, the spatial profile of the center-surround receptive fields of midget cells in primate retina or X cell in the cat retina can be well approximated by a linear operation with a Difference of Gaussian (DoG) function [32, 164]. The spatial profile of receptive fields of simple cells in V1 can be modeled as Gabor functions [22, 83]. The linearity property of receptive fields in these neurons has been extensively used in modeling single neuron responses [22, 87, 149, 177].

In what follows, we assume that the DSP of the neuron in Figure 3.1 accounting for the ith branch of the architecture, $i = 1, \cdots, M$, as shown in Figure 3.2, is a spatio-temporal linear filter, *i.e.*, the response of the filter subject to any input stimulus can be fully characterized by the first order (linear) kernel in its Volterra expansion. We denote the impulse response of the spatio-temporal filter by $h_1^i(x, y, t)$, $i = 1, \cdots, M$, where the subscript 1 indicates that the filter is the first order kernel in the Volterra expansion. The output $v^i(t)$ of the DSP is a filtered version of the visual scene and it is given by

$$v^i(t) = \int_{\mathbb{D}} h_1^i(x, y, t - s)u(x, y, s)dxdyds, i = 1, \cdots, M, \tag{3.6}$$

i.e., by integration in space and convolution in time.

Due to the presence of nonlinear effects in spike generators and gain control circuits in the early stages in the retina [7, 24, 160, 172], the response of neurons in the temporal domain may not be linear. In addition, some types of neurons, for example, Y cells in the cat retina and complex cells in V1 also exhibit highly nonlinear receptive fields [22, 45]. However, we consider in this chapter only linear DSPs. Nonlinear DSPs are accounted for in Section 6 and Section 7.

The output of the linear DSPs are provided as synaptic input currents to the BSGs discussed in Section 3.1.4.

3.1.4 Biophysical Spike Generators

Neurons fire action potentials when subject to proper stimuli. Conductance-based, biophysical models of spike generators capture the essence of electrical circuits underlying the dynamics of the membrane voltage of the neurons and their voltage-gated ion channels [81].

We consider here a class of BSGs whose dynamics can be expressed in compact form as

$$\frac{d\mathbf{x}^i}{dt} = \mathbf{f}^i(\mathbf{x}^i, I^i), \tag{3.7}$$

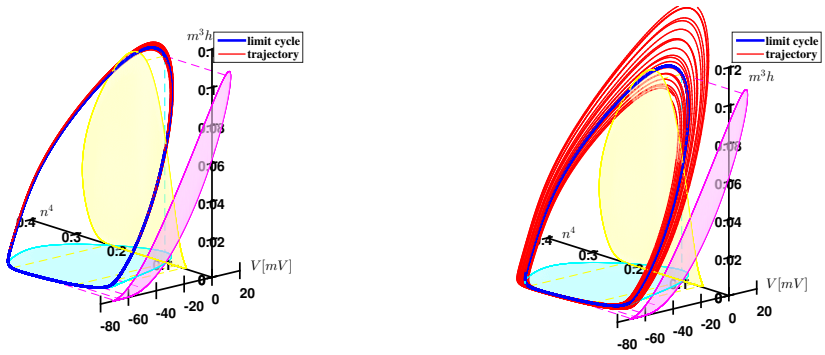
where $\mathbf{x}^i$ is a vector of dimension N^i, denoting the state (gating) variable of the neuron model, $\mathbf{f}^i$ is a vector function of the same dimension. As an example, $N^i = 4$ if neuron i is a Hodgkin-Huxley neuron (see Appendix A.1 for an detailed description of the Hodgkin-Huxley neuron model). We assume that (3.7) exhibits asymptotically stable limit cycles across a range of bias currents I^i. These include well known neuron models such as the Hodgkin-Huxley [76], Morris-Lecar and Connor-Stevens neurons [31, 135]. We refer to [60, 81] for a complete overview of biophysical models.

The output of the DSP $v^i(t)$ is additively coupled into the voltage equation (always the first equation) as in

$$\frac{d\mathbf{x}^i}{dt} = \mathbf{f}^i(\mathbf{x}^i, I^i) + \left[v^i(t), \mathbf{0}\right]^\mathsf{T}, \tag{3.8}$$

where $\mathbf{0}$ is a row vector of size $N^i - 1$ and $[\cdot]^\mathsf{T}$ denotes the transpose of $[\cdot]$.

The BSG model in (3.8) has been successfully used in formulating questions of encoding and decoding of current injections provided that input signals are weak [95]. In this case, the BSG model operates around a fixed limit cycle. Its dynamics can be large captured by linearization on the limit cycle. When signals are strong, however, the neural encoding problem is substantially more difficult to handle because the BSG model is highly non-linear. An example is shown in Figure 3.3. As the amplitude range of the DSP output varies widely, and this is certainly the case for encoding visual scenes, the trajectory of the neuron clearly deviates from the limit cycle. Consequently, the decoding methods based on small deviations from the limit cycle that was employed in the past [95] do not apply here.



(a) (b)

Figure 3.3: Trajectory of the Hodgkin-Huxley neuron in the phase space (V, n^4, m^3h), where V is the membrane voltage and, n^4 and m^3h are the fraction of open potassium and sodium channels, respectively. (a) The limit cycle (blue) and trajectory (red) of a Hodgkin-Huxley neuron with bias current I^i, subject to a weak stimulus. The trajectory stays close to the limit cycle. For better illustration, the projections of the limit cycle onto the planes V-m^3h, V-n^4 and n^4-m^3h are shown in half transparent magenta, cyan and yellow, respectively. (b) The limit cycle (blue) and trajectory (red) of the same neuron subject to a much stronger stimulus. The deviation of the trajectory from limit cycle is clearly visible. (Modified from [110] with permission, ©2014 IEEE)

3.2 Encoding as Generalized Sampling

In this section we show that encoding with a [DSP]-[BSG] neuron model can be interpreted as a projection of the input signal on a set of sampling functions. The projections are inner products in the Hilbert space $\mathcal{H}_3^s$. How can the sampling functions be determined? To answer this question we first derive the I/O relationship between the input and the occurrence time of spikes generated by a BSG. Characterizing the I/O of a BSG for weak signals is based on its PRC. For strong signals the conditional PRC is called upon. We then derive the I/O relationship between the input visual field (scene) and the multidimensional spike trains at the output of the neural circuit of Figure 3.2.

3.2.1 Characterizing the I/O of BSGs with a Single PRC

The response of the dynamical system (3.8) to time-varying inputs has been the subject of research with a broad range of approaches. Early studies focused on bifurcation analysis [81] and phase resetting [46]. Several reductionist approaches have been used to simplify the Hodgkin-Huxley model, *e.g.*, the Spike Response Model [65] or Volterra series [43]. Most recently, it has been suggested that ion channels of biological neurons can be modeled as memristors [28]. In contrast our methodology to characterize the I/O of BSGs is based on a family of PRCs.

PRCs are usually employed in neuroscience studies for describing the I/O behavior of biological spiking neurons. For example, the phase advance due to an input spike can be used to characterize the response of a neuron [19]. Prior studies often focussed on weakly coupled oscillators [47] and synchronization [73] with spike generators essentially operating on a fixed limit cycle. As we shall see in what follows, a family of PRCs is used here to describe the mapping of the continuous current input waveforms into spike sequences.

We assumed in Section 3.1.4 that the class of BSGs considered here have stable limit cycles (periodic orbits to (3.7)) across a range of bias currents I^i. The presence of limit cycles allows for analyzing the I/O relationship of BSGs in an one-dimensional phase space. The spiking period $T^i = T^i(I^i)$ associated with the limit cycle of neuron i depends on the bias current I^i.

For each asymptotically stable limit cycle, one can linearize the nonlinear differential equations (3.7) in the vicinity of the limit cycle. The linearized system becomes a Linear-Time-Varying (LTV) system in the phase space that can be characterized by a phase response curve (PRC). The PRC describes the phase advance when the BSG is subject to an infinitesimally small perturbation relative to the phase advance with no perturbation. We refer the readers to Appendix A.2 for a review of the PRCs of the Hodgkin-Huxley neuron model. For computing the PRCs the numerically efficient Malkin method [71, 81] is often used.

Since each PRC is associated with a limit cycle that is in turn induced by a constant current, we denote the family of PRCs of (3.7) with (constant) bias current I^i as

$$\boldsymbol{\psi}^i(t, I^i) = \left[\psi_1^i(t, I^i), \psi_2^i(t, I^i), \cdots, \psi_{N^i}^i(t, I^i)\right]^{\mathsf{T}}, \tag{3.9}$$

where ψ_n^i is the PRC associated with each state (gating) variable in (3.7), for all $i = 1, 2, ..., M$, and $n = 1, 2, ..., N^i$.

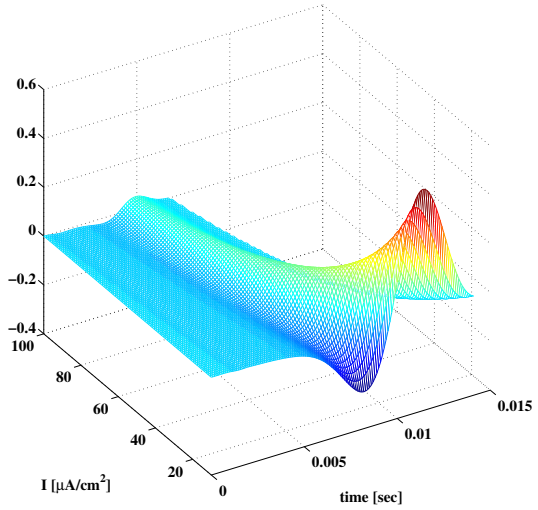


Figure 3.4: PRC manifold over the range of $I^i \in [10, 100]$ for an arbitrary neuron i. Only the PRC manifold of the Voltage equation of the Hodkgin-Huxley neuron i is shown. (From [110] with permission, ©2014 IEEE)

An example PRC manifold for an arbitrary Hodgkin-Huxley neuron i is shown in Figure 3.4. The manifold is obtained by plotting a family of PRCs parametrized by the bias current I^i. Provided that the inputs are small enough, the BSG dynamics of spike generation can be described in an one-dimensional phase space [95].

Definition 2. *A neuron whose spike times $(t_k^i), k \in \mathbb{Z}$, $i = 1, 2, ..., M$, verify the system of equations*

$$\int_{t_k^i}^{t_{k+1}^i} \psi_1^i(s - t_k^i + \tau^i(s - t_k^i, I^i), I^i) v^i(s) ds = T^i(I^i) - (t_{k+1}^i - t_k^i), \tag{3.10}$$

where for all $t \in [t_k^i, t_{k+1}^i]$

$$\frac{d}{dt}\tau^i(t - t_k^i, I^i) = \psi_1^i(t - t_k^i + \tau^i(t - t_k^i, I^i), I^i) v^i(s) ds, \tag{3.11}$$

with $\tau^i(0, I^i) = 0$, is called a Project-Integrate-and-Fire (PIF) neuron.

In equation (3.10) the PIF neuron projects $v^i(t)$, i.e., the input current onto the PRC. $\tau^i(t, I^i)$ in equation (3.11) denotes the phase deviation due to the perturbation on the right hand side. Note that the latter is the integrand in equation (3.10). Thus, the left hand side of equation (3.10) is the phase deviation due to the input signal measured between two consecutive spikes whereas the right hand side is the deviation of the interspike interval from the period of the limit cycle.

The BSG given by (3.8) and the PIF neuron in (3.10) and (3.11) are between two consecutive spikes, to a first order, I/O equivalent. The equivalence was carefully derived in [95]. The PIF neurons then replace the highly nonlinear BSGs in encoding weak inputs while keeping the I/O relationship unchanged. The PIF neuron described by the equations (3.10) and (3.11) was further simplified in [95] by simply setting the phase deviation in (3.10) to zero. The latter equation describes what is known as the reduced PIF neuron.

Definition 3. *The reduced PIF neuron [95] is given by the equations*

$$\int_{t_k^i}^{t_{k+1}^i} \psi_1^i(s - t_k^i, I^i) v^i(s) ds = T^i(I^i) - (t_{k+1}^i - t_k^i). \tag{3.12}$$

The reduced PIF neuron approximates the I/O relation of the BSGs well when inputs are weak, as the trajectory of the system does not significantly deviate from the original limit cycle.

3.2.2 Characterizing the I/O of BSGs with Conditional PRCs

The validity of the I/O equivalence between a BSG and a PIF neuron, as well as the approximation by reduced PIF neuron, relies on whether the trajectory of the BSG in the phase space is close to a limit cycle [95]. This is illustrated using a Hodgkin-Huxley neuron in Figure 3.3(a) for weak input current. When the strength of stimuli increases (Figure 3.3(b)), the trajectory does not always stay in the vicinity of the original limit cycle.

As the amplitude of DSP outputs varies over a wide range, the assumption of weak input does not hold for the visual encoding circuit. Stronger signals can be taken into account by considering the PRC manifold associated with a wide range of limit cycles [88]. To better approximate the trajectory of the system as it is projected onto the PRC manifold a spike-based estimate of the "current" limit cycle is needed. Interpreting it differently, at each point

in time the input signal is multiplied with one of the PRCs of the PRC manifold. The PRC in question, called the conditional PRC, is signal dependent. The estimated bias current, denoted by b_k^i, can be obtained from the inter-spike interval and the inverse of the function T^i, *i.e.*, $b_k^i = [T^i]^{-1}(t_{k+1}^i - t_k^i), k \in \mathbb{Z}$. The limit cycle and the conditional PRC of each interspike interval can be extracted accordingly. By estimating the limit cycle and hence its PRC through spike times, we have the following I/O relationship for each of the BSGs.

Definition 4. *The reduced PIF neuron with conditional PRC is given by the system of equations*

$$\int_{t_k^i}^{t_{k+1}^i} \psi_1^i(s - t_k^i, b_k^i)(v^i(s) - b_k^i + I_0^i)ds = 0, \tag{3.13}$$

where, $b_k^i = [T^i]^{-1}(t_{k+1}^i - t_k^i), k \in \mathbb{Z}$, *are estimated bias currents from the corresponding inter-spike intervals,* I_0^i *is the internal bias current,* $\psi_1^i(t, b_k^i)$ *is the conditional PRC.*

The conditional PRC in the definition of the reduced PIF neuron above is radically different from the PRC used in equation (3.12). Although, equation (3.13) is non-linear since b_k^i depends on v^i, it allows us to separate on short time windows the BSG input into a constant bias current and the fluctuations around it. This separation makes working with the conditional PRC more accurate for strong inputs. For example, we illustrate in Figure 3.5 that the estimated limit cycle (green dashed curve) is much closer to the actual trajectory than the limit cycle defined by the bias current I^i (blue solid curve).

The analysis in this section suggests that the process of encoding of strong signals with BSGs is substantially more complex simply because the non-linearity of the encoding circuit is more pronounced. The circuit is characterized by an input signal dependent PRC. In contrast simpler model encoders such as the ones based on the LIF neuron exhibit a single PRC that does not depend on the signal.

3.2.3 The t-Transform of the Encoding Circuit

With the I/O relationship of the BSG in mind, we now provide the overall I/O relationship for the neural circuit in Figure 3.2.

It is clear from Figure 3.5 that, due to a possibly large variation in the amplitude of the aggregated synaptic current, the PRCs of the BSGs are not always lying tight on the PRC

manifold (shown in Figure 3.4). Note that the reduced PIF neuron with conditional PRC is only valid when the BSG characterization is an integral part of the manifold of limit cycles during the inter-spike interval $\left[t_k^i, t_{k+1}^i\right]$. Therefore, we first have to empirically define a set of valid inter-spike intervals for which the reduced PIF neuron is a valid approximation.

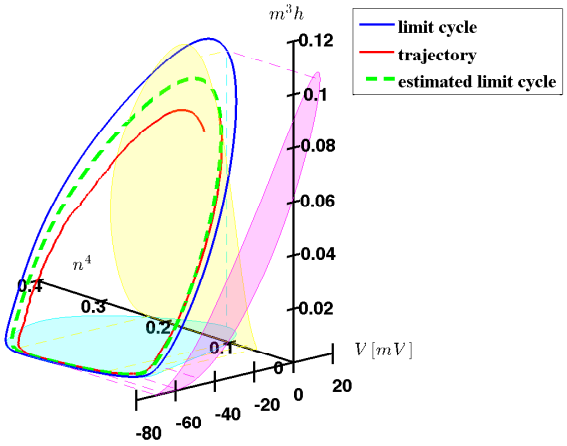


Figure 3.5: An example of the estimated limit cycle on which the conditional PRC is based upon. The red curve shows the trajectory between two consecutive spikes. The limit cycle defined by the bias current I^i (blue solid curve) is far away from the trajectory. The estimated limit cycle (green dashed curve) is much closer to the actual trajectory, particularly in the region around the resting potential, where the PRC takes larger values. (From [110] with permission, ©2014 IEEE)

Definition 5. *The set of indices k of spikes generated by BSG i that are valid for* (3.13) *is defined as*

$$\mathbb{I}^i = \{k | t_{k+1}^i - t_k^i \leq T_{max}\} \cap \{k | t_k^i - t_{k-1}^i \leq T_{max}\}, \tag{3.14}$$

where T_{max} is a predefined maximum value of interspike interval that is taken into account for encoding.

We denote the cardinality of $\mathbb{I}^i$ as $|\mathbb{I}^i|$. By Definition 5, we effectively identify three neuron states. The first state is a resting state; the neuron does not fire action potentials. The second state is the spiking state when the reduced PIF with conditional PRC is a valid approximation. The third state is in between the first two states while BSGs enter the

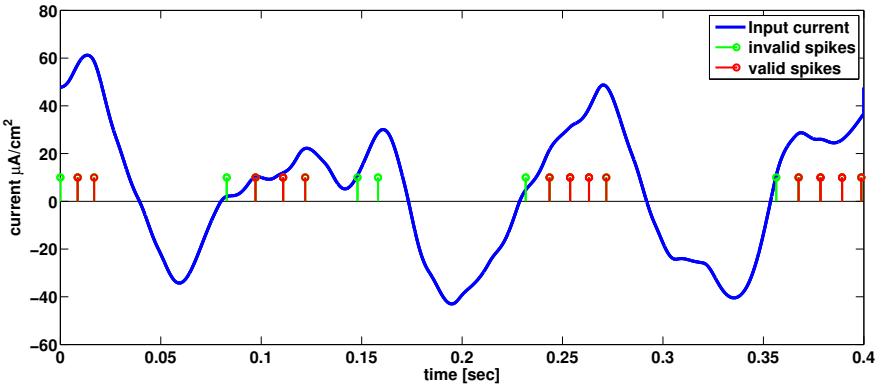


Figure 3.6: Example of input to a BSG and spikes generated. An Hodgkin-Huxley neuron was injected with a time-varying current (blue curve). 19 spikes were generated in response to the injected current. We only consider the t-transform on time intervals between spikes that are both labeled with red stems. The other inter-spike intervals are either too large (*e.g.*, between the 3rd and the 4th spike) and we deemed them as resting state, or they are in transition from resting state to spiking state, *i.e.*, they are in the ramp-up state (e.g., the spike interval between the 4th and 5th spike).

spiking state from resting state. We call this the ramp-up state. The second set in (3.14) excludes spikes that are generated in the ramp-up state. Those spikes empirically have larger peak amplitudes than spikes generated in the spiking state (see also Figure 3.8 for an example), and therefore, there is a clear separation between these two types of spikes.

We provide a simple example showing the set of valid spikes in encoding a signal with a Hodgkin-Huxley neuron. The input to the Hodgkin-Huxley neuron is shown in the Figure 3.6 (blue curve), while the spikes generated by the neuron in response to the input are indicated by stems. The spike intervals between any two red stems are deemed valid. All other spike intervals are discarded in formulating the encoding. We see that most of the discarded spike intervals correspond to a low input current.

Lemma 1. *The model of encoding in Figure 3.2 is given in operator form by*

$$\mathcal{T}_k^i u = q_k^i, k \in \mathbb{I}^i, i = 1, \cdots, M, \tag{3.15}$$

where $\mathcal{T}_k^i : \mathcal{H}_3^s \to \mathbb{R}, k \in \mathbb{I}^i, i = 1, \cdots, M$, *are bounded linear functionals,*

$$\mathcal{T}_k^i u = \int_{t_k^i}^{t_{k+1}^i} \varphi_k^i(t) \int_{\mathbb{D}} h_1^i(x, y, t-s) u(x, y, s) dx dy ds dt,$$

$$q_k^i = (b_k^i - I_0^i) \int_{t_k^i}^{t_{k+1}^i} \varphi_k^i(s) ds,$$

$$\varphi_k^i(t) = \psi_1^i(t - t_k^i, b_k^i).$$

(3.15) *is called the* **t-*transform*** *[106] of the neural circuit.*

Proof: (3.15) can be obtained by plugging (3.6) into (3.13). □

The t-transform describes the mapping of the continuous visual stimulus $u(x, y, t)$ into the spike sequences $(t_k^i), k \in \mathbb{I}^i, i = 1, 2, ..., M$. In Lemma 1, $\mathcal{T}_k^i$ is called the measurement operator, q_k^i is the value of the measurement. Lemma 1 suggests that the encoding of natural visual scenes by the neural circuit is merely generalized sampling in a Hilbert space. This is more apparent when we rewrite (3.15) in inner product form. By the Riesz representation theorem [10], there exists $\phi_k^i \in \mathcal{H}_3^s$ such that

$$\mathcal{T}_k^i u = \langle u, \phi_k^i \rangle, \forall u \in \mathcal{H}_3^s, \tag{3.16}$$

where, by the reproducing property (3.5),

$$\begin{aligned} &\phi_k^i(x, y, t) = \langle \phi_k^i, K_{xyt} \rangle = \mathcal{T}_k^i K_{xyt} \\ &= \int_{t_k^i}^{t_{k+1}^i} \varphi_k^i(t) \int_{\mathbb{D}} h_1^i(x', y', t-s) K(x', y', s; x, y, t) dx' dy' ds dt. \end{aligned}$$

(3.15) can then be expressed as

$$\langle u, \phi_k^i \rangle = q_k^i, k \in \mathbb{I}^i, i = 1, \cdots, M. \tag{3.17}$$

The above equation provides a simple interpretation to neural encoding: the encoded signal u is projected on a set of sampling functions ϕ_k^i. The result of this projection or measurement is q_k^i. Note that the sampling functions ϕ_k^i are not traditionally defined as fixed functions with shifts. Rather, they depend on the spike times generated for each input visual scene and thereby are input dependent.

The interpretation of the visual encoding neural circuit as performing generalized sampling is consistent with previous findings in encoding time-varying auditory stimuli with IAF neurons [96] and visual signals using phenomenological models such as LIF or TAF neurons [98, 112]. Therefore, the present result extends the generalized sampling framework to biological spiking neuron models. The complexity of the encoding process appears to be substantially higher.

3.3 Reconstruction of Natural Scenes from Spike Times

In [177] it was shown that visual scenes can be recovered, although coarsely, from recorded firing rates of neurons in the cat's LGN. The identified receptive fields were used in a linear decoder to reconstruct natural scenes from responses of the same ensemble of LGN neurons.

Is it possible then to reconstruct the encoded visual scenes from the observed multi-dimensional spike trains? This is a natural question that will shed light on the encoding process both in terms of decoding complexity but also as a measure of information that is contained in spikes. Since encoding can be interpreted as inner products or projections of visual scenes onto a set of sampling functions in a vector space, we expect that the signal can be recovered from these projections in a straightforward manner.

In Appendix 3.3.1 we will devise a decoding algorithm that recovers visual scenes encoded with the circuit shown in Figure 3.2. The applicability of our algorithm can be extended to encoding circuits based on ON-OFF neuron pairs as shown in Appendix 3.3.2.

3.3.1 Decoding Algorithm

In decoding, we assume that the encoding neural circuit is known and solve an inverse problem based on the I/O relationship and the observed output spike time occurrences.

Based on the generalized sampling result in Section 3.2, we adopt an approach similar to [108, 112] and formulate the reconstruction of the input visual scene as the smoothing splines optimization problem [12]

$$\hat{u} = \underset{u \in \mathcal{H}_3^s}{\operatorname{argmin}} \{\lambda \|u\|_{\mathcal{H}_3^s}^2 + \sum_{i=1}^{M} \sum_{k \in \mathbb{I}^i} (\mathcal{T}_k^i u - q_k^i)^2\} \tag{3.18}$$

corresponding to the measurement operators $\mathcal{T}_k^i$ and measurements q_k^i.

Theorem 1. *The reconstruction of the visual scenes encoded by a neural circuit with linear DSPs cascaded with BSGs, is*

$$\hat{u}(x,y,t) = \sum_{i=1}^{M} \sum_{k \in \mathbb{I}^i} c_k^i \phi_k^i, \tag{3.19}$$

where the c_k^i's are the solution of the system of linear equations

$$(\mathbf{\Phi} + \lambda \mathbf{I})\, \mathbf{c} = \mathbf{q}, \tag{3.20}$$

where $\mathbf{I}$ is the identity matrix of appropriate dimensions,

$$\mathbf{c} = \left[c_1^1, \cdots, c_{|\mathbb{I}^1|}^1, \cdots, \cdots, c_1^M, \cdots, c_{|\mathbb{I}^M|}^M \right]^{\mathsf{T}},$$

$$\mathbf{q} = \left[q_1^1, \cdots, q_{|\mathbb{I}^1|}^1, \cdots, \cdots, q_1^M, \cdots, q_{|\mathbb{I}^M|}^M \right]^{\mathsf{T}},$$

and $\mathbf{\Phi}$ is a block matrix of the form $\mathbf{\Phi} = [\mathbf{\Phi}^{ij}]$, with

$$[\mathbf{\Phi}^{ij}]_{kl} = \langle \phi_k^i, \phi_l^j \rangle, i, j = 1, \cdots, M, k \in \mathbb{I}^i, l \in \mathbb{I}^j.$$

Proof: The proof of Theorem 1 can be found in Appendix B.2.

Remark 2. *The reconstruction based on Theorem 1 provides a solution to the smoothing spline problem in a subspace of $\mathcal{H}_3^s$ generated by the span of all sampling functions $(\phi_k^i), k \in \mathbb{I}^i, i = 1, 2, ..., M$. If the span equals $\mathcal{H}_3^s$, then the visual scene can be recovered faithfully under noiseless conditions [112]. This requires the sampling functions to form a frame in $\mathcal{H}_3^s$. This can be guaranteed by carefully choosing a set of DSPs and acquiring enough measurements (spikes) over a certain time period [98]. As $\mathcal{H}_3^s$ considered here is finite dimensional, a necessary condition for $(\phi_k^i), k \in \mathbb{I}^i, i = 1, 2..., M$, to span $\mathcal{H}_3^s$ is that the total number of spikes is greater than $(2L_x + 1)(2L_y + 1)(2L_t + 1)$, the dimension of $\mathcal{H}_3^s$. In addition, a minimum of $(2L_x+1)(2L_y+1)$ neurons are required as this is the dimension in visual space. In the limit when $L_x, L_y, L_t \to \infty$, the visual scenes converge to bandlimited signals and the condition for recovery becomes a Nyquist-type rate condition on spike times and frame condition on the bank of DSPs in space [98].*

As already mentioned, algorithms for decoding visual scenes from the spike times generated by the Video TEMs are referred to as Video TDMs [98]. Since the RKHS we considered here is finite dimensional, the form of reconstruction can also be equivalently formulated with basis functions instead of sampling functions [112]. However, Theorem 1 applies generally to infinite dimensional RHKSs by observing that the reconstruction takes place in the subspace generated by the sampling functions.

We summarize the decoding procedure of the visual stimulus encoded by a population of biophysical spike generators in Figure 3.7. First, for each valid inter-spike interval, the conditional PRC is estimated. Measurement operators and their measurements can then be evaluated, and the entries of the $\mathbf{\Phi}$ matrix filled. The linear system is solved and the solution $\mathbf{c}$ drives the reconstruction of the visual scene.

The decoding algorithm serves as a probe for quantitatively estimating how well the visual stimuli are represented by the multi-dimensional spike trains. The reconstruction of the visual signal can be intuitively compared to the original in the input space. A variety of quantitative assessment methods are available for evaluating the faithfulness of representation in the space of natural visual scenes [27].

The conditions on the reconstruction algorithm have the following implications. First, they impose a constraint on the size of the encoding neural circuit. Second, there is an upper bound on the maximum bandwidth that the neural circuit can support. Taken together, the design of the encoding neural circuit becomes tractable. Third, they suggest that the use of a proper model of the input space is critical to the understanding of neural encoding of visual scenes, as it has always been in studies of sampling theorems [185].

3.3.2 Extensions: ON-OFF Pairs and Feedback DSP

Responses of biophysical neurons to positive and negative inputs are asymmetric. For the deterministic model, neurons generate spikes only when either the overall input current is larger than a certain value, or there is a significant and fast increase of the input when the neuron is in its resting state.

Moreover, since the input to each of the biophysical spike generators may be large in amplitude while being negative, we introduced a small modification in the differential equa-

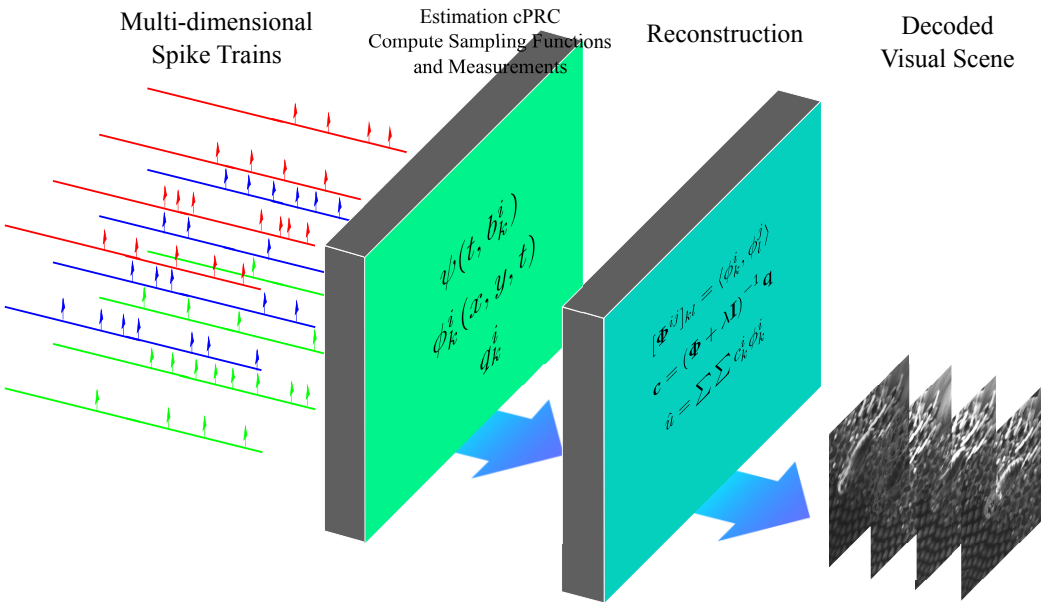


Figure 3.7: Schematic diagram of Video Time Decoding Machine.

tions (3.8) to avoid driving the membrane voltage to unrealistically low values. This goal is achieved by forcing the membrane voltage to be larger than a certain value $V \geq V_{low}$. In model simulations we set the value of V_{low} to -80 [mV], a value in the range of reversal potentials of many inhibitory neurotransmitters [48]. All these and other observations suggest the need for alternative BSG models for encoding to ensure that the reconstruction conditions hold.

Since BSGs do not fire spikes if their inputs are below a certain nominal value, two strategies can be used to remedy the lack of spikes whenever input currents are negative. For the first, an elevated bias current can ensure that the inputs are always in the range that can generate spikes. For the second, two neurons can be employed, with one encoding the input and the other encoding the inverted input.

An example of the latter strategy is provided in Figure 3.8 with two BSGs, one encoding the input current (in blue) and the other encoding the negative (inverted) input current (in red). The membrane voltages are shown for each BSG, in blue and red, respectively. Spike times are indicated by stems. This suggests that the two neurons can encode an input with an alternating spiking pattern.

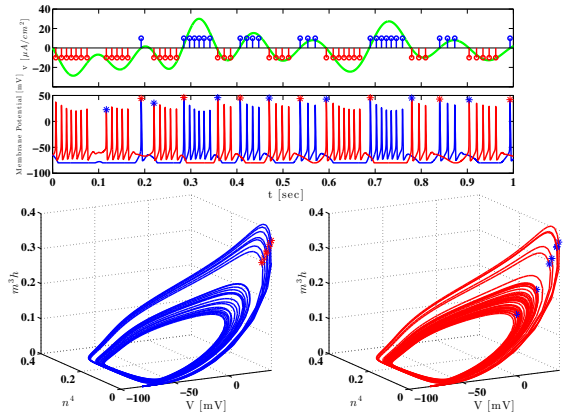


Figure 3.8: An example of two BSGs encoding the signal $v(t)$ (green) and $-v(t)$, respectively. BSG 1 encodes v; its spike times, membrane potential and phase plane trajectory are shown in blue in the top, middle and bottom left plots, respectively. BSG 2 encodes $-v$; its spike times, membrane potential and phase plane trajectory are shown in red in the top, middle and bottom right plots, respectively. The onset spikes are highlighted by a star of opposite color to the respective membrane potential and trajectory. (From [110] with permission, ©2014 IEEE)

Biologically, the concurrent encoding of the input and the inverted input corresponds to ON-OFF DSPs. For example, neurons in the retina with center-surround type receptive fields come in two forms. They have a receptive field either with ON center and OFF surround, or with OFF center and ON surround [93]. Encoding with ON and OFF type receptive fields has been the subject of information theoretic analysis and natural image statistics investigations [92, 158].

To build a biologically relevant encoder, we constructed a neural circuit by adding both ON and OFF type DSPs in cascade to the BSGs with $h_1^j(x,y,t) = -h_1^i(x,y,t)$, as shown in Figure 3.9. Furthermore, we included feedback loops between the ON and OFF BSGs. BSG i is associated with an ON DSP and BSG j is associated with an OFF DSP. Both DSPs have the same spatial center. Feedback currents are modeled as the linear readouts of the impulse response of feedback filters at spike time, *i.e.*, $h_1^{ij}(t - t_k^i)$ at spike time t_k^i.

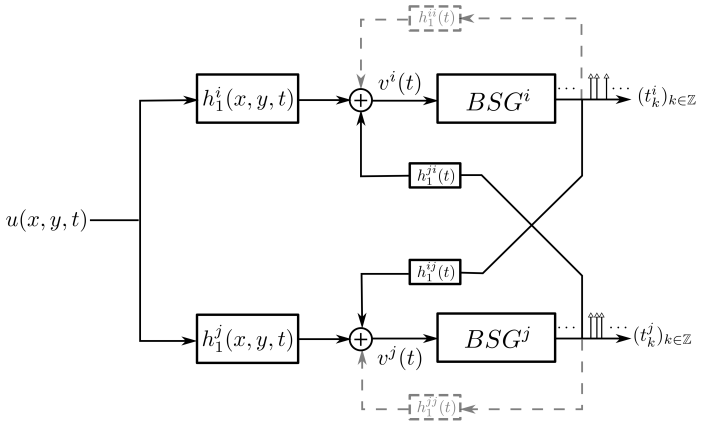


Figure 3.9: Schematic diagram of a pair of neurons with self- and cross-feedback. For the ON and OFF neuron pair, $h_1^j(x,y,t) = -h_1^i(x,y,t)$ (change of sign). (From [110] with permission, ©2014 IEEE)

ON-OFF pairs with self- and cross-feedback have been successful in modeling neurons in early visual systems [112, 149, 150]. The cross-feedback models the coupling between a pair of neurons. The coupling often arises from the lateral interaction among a population of neurons [13, 153, 194].

We consider the self-feedback optional since self-feedback loops are usually introduced for

phenomenological spiking models (*e.g.*, Poisson spiking, IAF neurons) to artificially create refractoriness (a second spike cannot be generated immediately after the first spike) [87, 149]. However, the BSGs used here have a refractory period built in. Nevertheless, the self-feedback can still be used to model any effect that is directly a consequence of spikes generated by the neuron itself [48].

Example 1. *We provide here an example to illustrate the use of an ON-OFF pair and feedback in encoding. Figure 3.10 shows the spikes generated in 5 different scenarios. Input* $v^i(t)$ *is shown in green.*

The top row repeats the top plot in Figure 3.8. A strong signal (maximum amplitude is 30 $[\mu A/cm^2]$*) is encoded by two Hodgkin-Huxley neurons in an ON-OFF configuration, with bias current of the ON neuron* $I^1 = 0$ *and that of the OFF neuron* $I^2 = 0$*.* 64 *spikes are generated by the two neurons.*

The second row shows the same signal encoded by only a single Hodgkin-Huxley neuron with elevated bias $I^1 = 40$*.* 103 *spikes are generated, a number that is substantially larger than in the ON-OFF configurations. It can be seen that, compared with a strong bias current, the ON-OFF configurations is effective in reducing the number of spikes while generating spikes over the entire input domain. This is important to guarantee decoding as discussed in Section 3.3.1.*

The third row shows the encoding of a much weaker signal (maximum amplitude is 6 $[\mu A/cm^2]$*). Since the signal is effectively below the minimum amplitude that can elicit a spike, no spikes are generated by the Hodgkin-Huxley neuron.*

Bias current $I^1 = I^2 = 10$ *is used for both ON and OFF neurons in the fourth row. Still, only the OFF neuron fires spikes during some time intervals.*

In the fifth row, transient, inhibitory cross-feedback is introduced between the ON-OFF pair, with bias current $I^1 = I^2 = 10$ *for the two neurons. The introduction of cross-feedback restores the alternating spiking between the two neurons. Spikes from both BSGs cover the input domain.*

The feedback loops can also be generalized to be between any two neurons in the encoding circuit rather than between a pair of ON and OFF neurons since the strict association of an

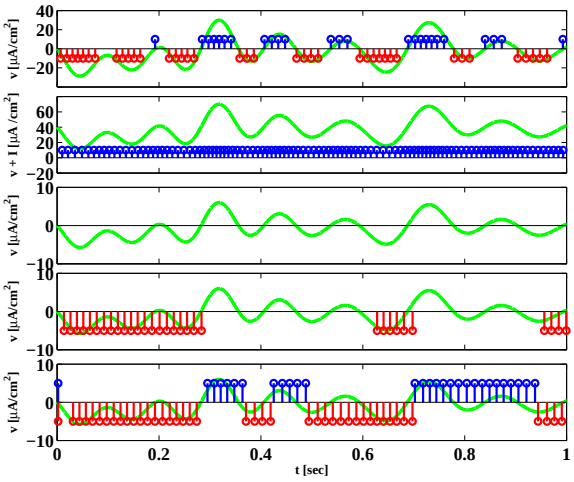


Figure 3.10: Examples of encoding time varying signals with ON-OFF neuron pairs and feedback. Green curves show the input signals. Blue stems show the spike times of the ON neuron and the red stems show the spike times of the OFF neuron. Each row corresponds to a different setting in the bias current and feedback. (see Example 1 for more details.) (From [110] with permission, ©2014 IEEE)

ON and an OFF BSG may be weak.

With self- and cross-feedback and combining with (3.6), the input currents to the BSGs amount to

$$\begin{aligned} v^i(t) = &\int_{\mathbb{D}} h_1^i(x,y,t-s)u(x,y,s)dxdyds \\ &+ \sum_{j=1}^{M}\sum_{l\in\mathbb{Z}} h_1^{ji}(t-t_l^j), i = 1,\cdots,M. \end{aligned} \tag{3.21}$$

3.4 Algorithmic Considerations in Reconstruction

The huge amount of spikes generated by a massively parallel neural circuit when encoding natural visual scenes make it difficult to solve (3.20) for the entire set of spikes. We introduce in this section two algorithms that enables the reconstruction of visual stimuli with large screens on a GPU cluster. We first show that the reconstruction problem can be solved by a recurrent neural network in Appendix 3.4.1. In Appendix 3.4.2, we describe a volume stitching algorithm in order to reconstruct a visual scene encoded by a massively parallel

neural circuit.

3.4.1 Reconstruction Using Recurrent Neural Network

In order to solve (3.20), either a matrix inversion or Moore-Penrose pseudo-inverse is often employed. Evaluating the inverse and pseudo-inverse are, however, computationally demanding operations. To fully preserve the information contained in the visual scene, a very large number of spikes is required, and hence the order of $\mathbf{\Phi}$ may be exceedingly large. It is usually not practical to invert $\mathbf{\Phi}$ directly.

Linear equations or optimization problems using recurrent neural networks based on differential equations have been extensively studied in the last two decades [29]. These networks contain neuron-like operations, such as the weighted sum and integration operations that provide a more plausible solution to the reconstruction problem. Here, we observe that the matrix $\mathbf{\Phi}$ in the reconstruction algorithm is positive-semidefinite. The least squares solution to (3.20) can be approximated by the stationary point of a recurrent neural network.

Theorem 2. *The solution to the spline interpolation problem* (3.18) *is*

$$\hat{u}(x,y,t) = \sum_{i=1}^{M} \sum_{k \in \mathbb{I}^i} c_k^i \phi_k^i, \tag{3.22}$$

where $\mathbf{c}$ *is the stationary point of the system of differential equations*

$$\frac{d\mathbf{c}}{dt} = \alpha \left(\mathbf{q} - (\mathbf{\Phi} + \lambda \mathbf{I})\mathbf{c} \right), \tag{3.23}$$

with initial condition $\mathbf{c}(0) = \mathbf{0}$ *and* $\alpha > 0$.

Proof: Let $\mathbf{G} = \mathbf{\Phi} + \lambda \mathbf{I}$. For the energy function $E(\mathbf{c}) = \frac{1}{2}\|\mathbf{Gc} - \mathbf{q}\|^2 \geq 0$, we have

$$\frac{dE}{dt} = \sum_{i=1}^{M} \sum_{k \in \mathbb{I}^i} \frac{\partial E}{\partial c_k^i} \frac{dc_k^i}{dt} = (\nabla E)^{\mathsf{T}} \frac{d\mathbf{c}}{dt} = -(\mathbf{Gc} - \mathbf{q})^{\mathsf{T}} \mathbf{G} (\mathbf{Gc} - \mathbf{q}) \leq 0.$$

The last inequality is due to the fact that $\mathbf{G}$ is positive-semidefinite. Therefore, the time derivative of the energy function is monotonically decreasing, and the equalibrium condition, $\frac{dE}{dt} = 0$, is satisfied if and only if $\mathbf{Gc} - \mathbf{q} = 0$ or $\mathbf{G}^{\mathsf{T}}(\mathbf{Gc} - \mathbf{q}) = 0$. In other words, the solution to equation (3.23) is guaranteed to converge to a stationary point $\mathbf{c}$ that satisfies

either exactly $\mathbf{Gc} = \mathbf{q}$, or $\mathbf{c} = (\mathbf{G}^\mathsf{T}\mathbf{G})^+\mathbf{G}^\mathsf{T}\mathbf{q} = \mathbf{G}^+\mathbf{q}$, where $\mathbf{G}^+$ denotes the Moore-Penrose pseudo-inverse of $\mathbf{G}$. □

Equation (3.23) can easily be implemented by a circuit consisting of neural components such as integrators and adders; the Video TDM can be realized as the diagram shown in Figure 3.11. Note that the above system of differential equations may converge slowly. In such cases the circuit simulation can be stopped while still guaranteeing a high quality reconstruction of the visual stimuli (see also Appendix 3.5).

The recurrent neural circuit in Figure 3.11 is massively parallel. It consists of $n = \sum_{i=1}^{M} |\mathbb{I}^i|$ parallel branches. This number can be very large since it represents the total number of spikes generated by the encoder. In each branch, only two simple components are required - a multiply/add unit and an integrator. Such simple circuits can be effectively realized on a high performance computer.

The original method of evaluating the pseudo-inverse, on the one hand, is typically based on an elegant mathematical treatment using SVD. It requires additional workspace in memory that limits the problem size that can be solved, and although scaling the algorithm to multiple computing devices is possible, it is highly nontrivial. The recurrent neural network approach, on the other hand, is rather straightforward to both scale up and scale to multiple computing devices, such as a cluster of GPUs. It does not require extra workspace so that the entire memory resource can efficiently be utilized.

3.4.2 Volume Stitching in Reconstruction

Even if the stimulus reconstruction is performed on a large scale GPU cluster it is still necessary, due to the massive number of neurons and massive number of spikes generated in encoding for a large aperture stimulus, to divide the stimulus into smaller volumes and focus the reconstruction on each volume. After all volumes are reconstructed, they can be stitched together using a stitching algorithm following a procedure similar to the one in [111]. We now describe how each stimulus segment is reconstructed and provide the stitching algorithm for the complete recovery of the visual stimulus.

We first divide the stimulus into pieces of fixed size, overlapping volumes, as illustrated

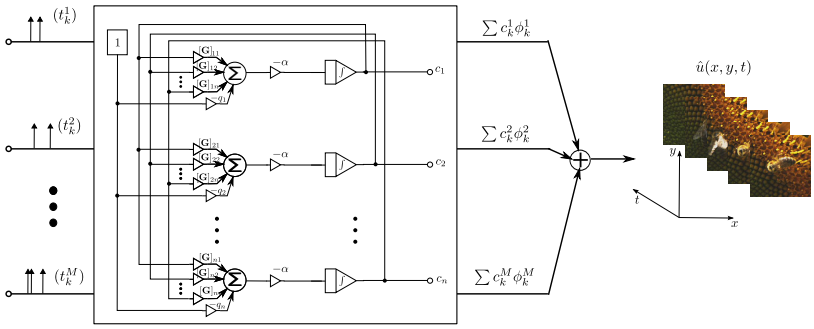


Figure 3.11: Block diagram of the Video TDM implemented using recurrent neural networks. The recurrent neural network is shown in the square box. (Modified from [108] with permission, ©2012 Elsevier)

in Figure 3.12. We denote the length of each volume in x, y and t direction as J_x, J_y and J_t, respectively. The length of the overlapping part of two adjacent volumes in the x, y and t directions is denoted by O_x, O_y and O_t, respectively, with $2O_x < J_x, 2O_y < J_y, 2O_t < J_t$. We define by $(V_{k,l,m}), k, l, m \in \mathbb{Z}$, the volume segment localized in

$$(k(J_x - O_x), (k+1)(J_x - O_x) + O_x] \times (l(J_y - O_y), (l+1)(J_y - O_y) + O_y] \\ \times (m(J_t - O_t), (m+1)(J_t - O_t) + O_t],$$

and by $\hat{u}_{k,l,m}(x, y, t)$ the stimulus reconstruction based on spikes localized in the volume segment $V_{k,l,m}$. $\hat{u}_{k,l,m}$ is obtained by the decoding procedure described in Section 3.3.1 and only takes into account the spikes localized in the volume $V_{k,l,m}$ that satisfy the following conditions (i) they are generated by neurons whose receptive fields are centered inside the spatial domain of $V_{k,l,m}$ and (ii) the spike times are inside the temporal domain of $V_{k,l,m}$ or are exactly the closest spikes before or after the temporal segment.

The stimulus reconstructions of the individual volume segments are stitched together with a simple shifting windows algorithm. We define the windows

$$w_{k,l,m} = w_k^x \cdot w_l^y \cdot w_m^t, \tag{3.24}$$

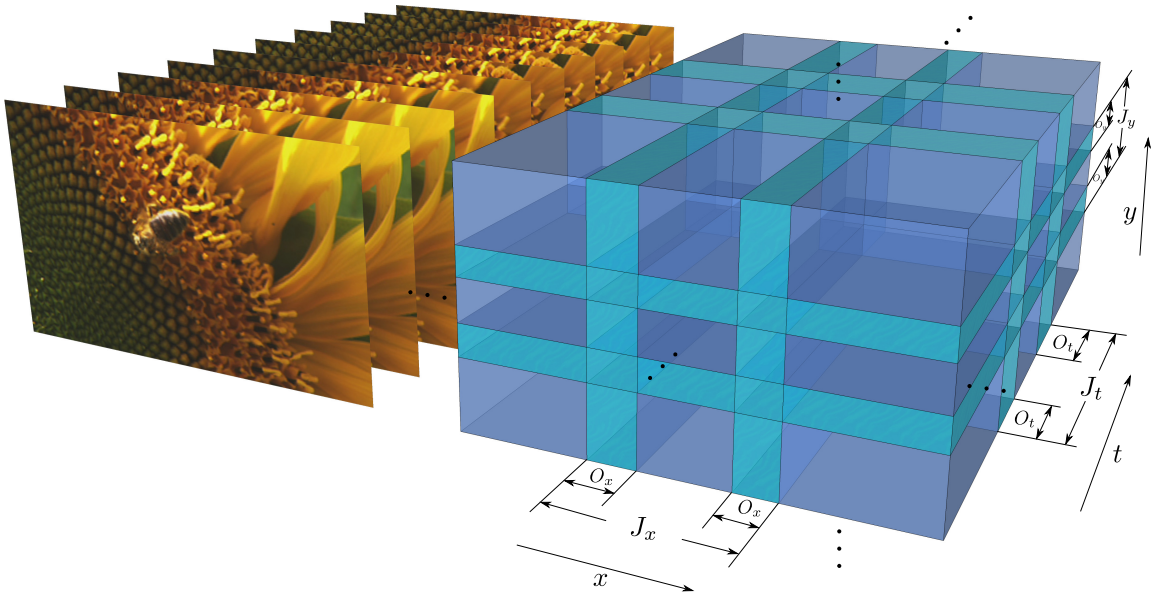


Figure 3.12: (left) Space-time natural video sequence. (right) Division of the space-time video sequence into fixed sized, overlapping volume segments. The brighter blue color indicates the overlapping adjacent volume segments. (Modified from [108] with permission, ©2012 Elsevier)

where

$$w_k^x = \begin{cases} 0, & x \notin (k(J_x - O_x), k(J_x - O_x) + J_x], \\ \theta_k^x, & x \in (k(J_x - O_x), k(J_x - O_x) + O_x], \\ 1, & x \in (k(J_x - O_x) + O_x, (k+1)(J_x - O_x)], \\ 1 - \theta_{k+1}^x, & x \in ((k+1)(J_x - O_x), (k+1)(J_x - O_x) + O_x], \end{cases} \tag{3.25}$$

with θ_k^x is an appropriately chosen function, for example,

$$\theta_k^x = \sin^2\left(\frac{\pi}{2} \cdot \frac{x - k(J_x - O_x)}{O_x}\right), \tag{3.26}$$

The functions w_l^y and w_m^t are similarly defined. It is easy to see that the defined window functions form a partition of unity.

The overall visual stimulus reconstruction is therefore given by

$$\hat{u}(x, y, t) = \sum_{k,l,m \in \mathbb{Z}} w_{k,l,m} \hat{u}_{k,l,m}(x, y, t). \tag{3.27}$$

Since a fixed sized volume is used for stitching, the shape of the window functions are all the same and they can be pre-computed. Thus, the overall visual stimulus reconstruction can be obtained by reconstructing the visual stimulus in each volume segment, multiplying these with the stored window functions, and then stitching the latter in an 'overlap-add' fashion.

3.5 Examples

We provide here an example of encoding of natural visual scenes using the neural circuit depicted in Figure 3.2 that models the encoding by a population of RGCs in the primate retina. We then decode the video from the spike times. All simulations were performed on a cluster of 55 Nvidia Kepler GPUs.

The encoded visual scene is a 2 second, 360pixel×640pixel natural video. The video has been spatially pre-filtered to ensure that it is bandlimited to a certain effective bandwidth. This is a standard procedure in (amplitude) sampling to avoid aliasing. This step can be considered to model the low-pass spatial filtering by the eye's optical system.

The color of the visual scene is represented in the RGB color space. We performed encoding and decoding on each channels separately before combining them for visualizing the result. The encoding settings for all the channels are the same. In the following, we describe the encoding circuit for encoding one channel.

The encoding neural circuit consists of $377,010$ BSGs governed by Hodgkin-Huxley equations with bias current 10 $[\mu A/cm^2]$. The bias current is still necessary to avoid the dead zone in the ON-OFF setting. Its small value is in the low range of bias currents that can induce periodic oscillations (7-150 $[\mu A/cm^2]$), and amounts to about 6% of the maximum amplitude variation of current inputs (see also Example 1). The DSP associated with each BSG is defined, for simplicity, only spatially with a center-surround, DoG filter, where half of the BSGs has ON-center DSPs and the rest have OFF-center DSPs. The mother function of the impulse response of the DSP is defined as

$$h_1^i(x, y, t) = h_1^i(x, y)\delta(t),$$

where $\delta(t)$ is the Dirac delta function and

$$h_1^i(x,y) = \frac{1}{\sigma_c^2}\exp\left(-\frac{x^2+y^2}{2\sigma_c^2}\right) - C\frac{1}{\sigma_s^2}\exp\left(-\frac{x^2+y^2}{2\sigma_s^2}\right)$$

with σ_c^2, σ_s^2 as the variance for center and surround Gaussian functions and C a scaling factor [32, 198].

A total of 5 octave scales of dilations were used. For each dilation scale, the centers of ON DSPs tile the visual space with a hexagonal array, where the distance between the centers of two neighboring DSPs is set to $2\sigma_c$, a distance observed in the primate retina [61]. Same hexagonal array is used for OFF DSPs at the same scale. The parameters of the DoG filters take into account experimental observations in primate retina [32], where $\sigma_c = 7\sigma_s$ and $C = 0.9$.

Additionally, a pair of ON and OFF BSGs with the same dilation scale and DSP centers provide cross-feedback. Self-feedback loops are not used in the simulation.

A total of $28,659,941$; $29,184,254$ and $31,589,289$ spikes were generated for encoding red, green and blue channels, respectively. These numbers roughly correspond to 40 [spikes/sec] for each BSG. Encoding of every 1 second of monochromatic video using 16 GPUs took 75 seconds. Spike timings were quantized to 1 [μsec]. The temporal bandwidth of the input video was assumed to be 10 [Hz] in decoding. Therefore, the average number of spikes generated by the encoder is about 4 times the Nyquist-rate of the video (considering ON and OFF fire together at 80 [spikes/sec] and the Nyquist-rate is 20 [samples/sec]).). The relatively high ratio is primarily due to the intrinsic firing rate of the Hodgkin-Huxley neurons that is in the range of 60-140 [spikes/sec]. However, at this average rate the reconstruction algorithm also performed well for 20 [Hz] temporal bandwidth (data not shown).

Decoding was done by reconstructing a block volume each time and by employing volume stitching as described in Section 3.4.2. The $360\times640\times2$ [pixel×pixel×sec] video was divided into $144 \times 140 \times 0.15$ [pixel×pixel×sec] blocks for decoding a total of 360 blocks. In other words, the parameters defining the windows $V_{k,l,m}$ were $J_x = 140, J_y = 144, J_t = 0.15$ and $O_x = 40, O_y = 36, O_t = 0.05$. We chose the parameters of $\mathcal{H}_3^s$ as $L_x = L_y = 40$, $L_t = 8$, $\Omega_x = \Omega_y = 0.25$, $\Omega_t = 10 \cdot 2\pi$. We also used a space with all the spectral lines inside of a circular area instead of a square area as defined in (3.1), *i.e.*, we restricted l_x and l_y to be

in the set $\{(l_x, l_y)|l_x^2/L_x^2 + l_y^2/L_y^2 \leq 1\}$. This way, the bandwidth it covered in an isotropic fashion. T_{max} in Definition 5 was set to 14.6 [msec], the period corresponding to the bias current 10 $[\mu A/cm^2]$. About $200,000$ measurements were used in the reconstruction of each block - about 2.4 times the dimension of the space $(85, 425)$. Decoding of each second of the full-screen video took 9 hours on 55 GPUs. We show the video in Figure 3.13 the original and reconstructed video, along with the error. The SNR for recovered R, G and B channels are 35.9 [dB], 34.2 [dB] and 30.9 [dB], respectively. Mean SSIM index values [190] are 0.995, 0.996, 0.998 respectively.

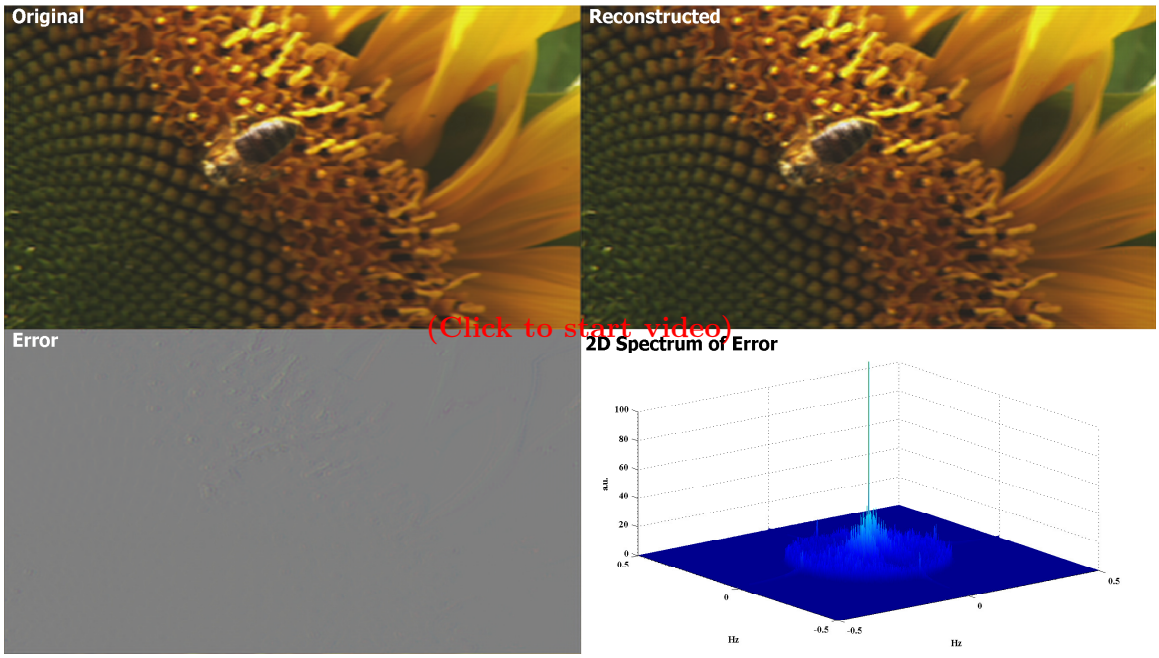


Figure 3.13: Example of reconstructed visual scene. (upper left) A snapshot of the original visual stimulus. (upper right) A snapshot of the reconstruction at the same time instant as the original. (bottom left) Error between original and reconstruction. (bottom right) The 2D spectrum of the R error component. (From [110] with permission, ©2014 IEEE)

In Remark 2, it is suggested that the reconstruction quality is also related to the number of measurements generated in encoding relative to the dimension of $\mathcal{H}_3^s$. We next show this relation using a few different encoding configurations. The visual scene is a 90pixel×160pixel subdomain of the previous scene. The key parameters of $\mathcal{H}_3^s$ are chosen as $L_x = 22, L_y =$

$39, L_t = 4, \Omega_x = \Omega_y = 0.244 \cdot 2\pi, \Omega_t = 10 \cdot 2\pi$. The spatial dimension of the space is $2,679$ while the number of neurons required for reconstruction is twice the spatial dimension since the circuit involves both ON and OFF BSGs. First, we gradually increase the distances between two neighboring DSP centers on the hexagonal array. This effectively reduces the number of neurons that encode the visual scene. Note that the exact theoretical bound on the minimum number of neurons required can be more readily obtained with randomly generated DSPs. Here, in the interest of biological relevance we still used center-surround linear DSPs. The reconstruction quality is shown the video in Figure 3.14 as a function of the number of neurons used in encoding, with a snapshot of the reconstruction for each configuration. The SNR values are interpolated with a cubic spline. The distances between two neighboring DSP centers with respect to σ_c, the parameter of the center Gaussian function, are indicated on top of each snapshot. It can be seen that as the grid becomes denser and more neurons are used to encode the visual scene, the reconstruction quality increases. This suggest that the information in the visual scene is represented increasingly better as more neurons are involved in encoding, and hence more spikes are generated. Moreover, as the hexagon grid reaches the density that corresponds to the neighboring distance of $2\sigma_c$, the reconstruction performance saturates. This indicates the spikes generated by the encoding neural circuit fully represent the visual scenes at this biologically observed value. This example highlights the spatial aspect of the reconstruction condition.

To illustrate the temporal aspect of the reconstruction condition, we use a fixed neural circuit to encode the visual scenes with increasing temporal bandwidth. The increase of the temporal bandwidth is achieved by using the same video sequence but presenting it to the encoding circuit at a faster speed. Since the neural circuit is fixed, it can generate a limited amount of spikes per unit time. Therefore, as temporal bandwidth of the visual scene increases, the encoder fails to encode the faster changes in the visual scene. We encoded and decoded the visual scene at original speed (10 [Hz] effective bandwidth), 2 times, 4 times and 6 times speed up. The resulting reconstruction quality is shown as a function of the maximum effective bandwidth in the video in Figure 3.15, with the reconstructions for different bandwidth. The reconstruction of the visual scenes at different speeds is played back with an appropriate slow down to allow comparison. As expected, at the highest bandwidth,

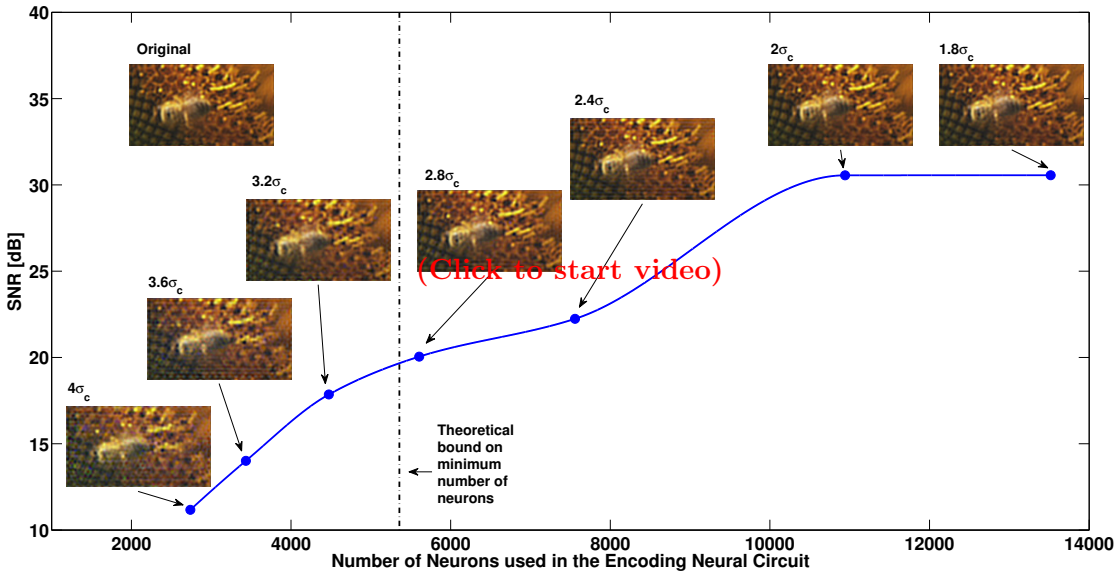


Figure 3.14: Reconstruction quality as a function of the number of neurons used in encoding. Neural circuits with different density of neurons are used to encode the same visual scene. The density is parametrized by the distance between centers of neighboring DSPs on hexagon grids relative to the parameter σ_c of the center Gaussian of the center-surround DSP. The distance values are shown on top of each snapshot of the reconstruction. (From [110] with permission, ©2014 IEEE)

the decoder cannot reconstruct as well the visual scenes as at the lowest bandwidth. This example highlights the temporal aspect of the reconstruction condition.

3.6 Summary

The neural circuit architecture investigated in this chapter models the encoding of natural visual scenes into multidimensional spike trains. The architecture accommodates, in addition to analytically tractable spike generators such as the IAF neuron, biophysical spike generators such as Hodgkin-Huxley neuron.

The difficulty of understanding the visual encoding process in circuits built with BSGs lies in the fact that their characteristic non-linearity is substantially more pronounced in

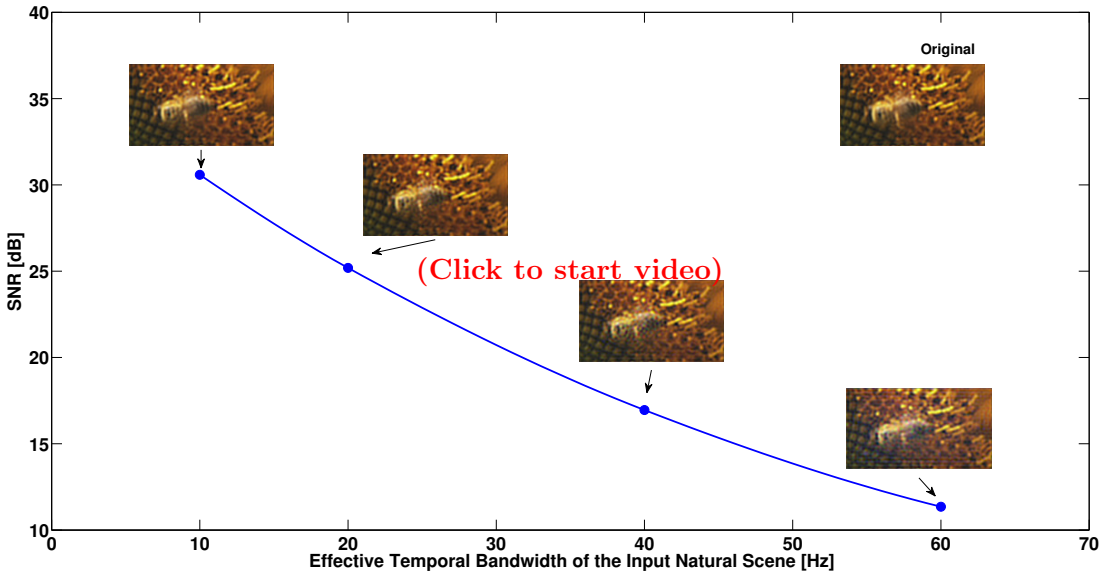


Figure 3.15: Reconstruction quality as a function of the the maximum temporal bandwidth in the visual scene. The same circuit encodes the visual scene with different bandwidth. Bandwidth is simulated by presenting to the encoding circuit a speed-up version of the same visual scene. As the temporal bandwidth increases, the fixed neural circuit with a maximum spiking rate limit cannot fully encode the faster changing visual scenes. (From [110] with permission, ©2014 IEEE)

encoding visual scenes. In this regime the assumption previously made that BSGs operate on a single limit cycle is no longer valid. We extended the encoding model and introduced for its analysis a manifold of PRCs; for decoding we used signal dependent conditional PRCs. This suggests that the complexity of the encoding process with BSGs is substantially higher than with simplified models.

Consistent with previous results of visual encoding with simplified neuron models, spikes generated by a neural circuit with linear DSPs cascaded with BSGs can be viewed as the result of generalized sampling in Hilbert spaces. We formulated the decoding of visual scenes as a smoothing spline optimization problem. For natural visual scenes modeled as spatio-temporal bandlimited signals, a Nyquist-type rate condition can be imposed on the number

of neurons and the number of spikes to guarantee the faithful reconstruction of the signal.

We then extended the encoding neural circuits incorporating ON-OFF type DSPs (receptive fields) and self- and cross-feedback. Combined with conditional PRCs, the biologically relevant ON-OFF circuit configuration relaxes or even eliminates the need for bias currents in the BSGs. This has important implications on the power budget of these circuit architectures when implemented in silicon.

Finally, we demonstrated the massive encoding and decoding of natural visual scenes by simulation on GPUs. We investigated the performance in recovery in the Nyquist region and showed graceful degradation of signal recovery. We also investigated the sensitivity of signal reconstruction with respect to bandwidth demonstrating thereby the failure of recovery when the bandwidth exceeds what a particular encoding neural circuit can support.

Chapter 4

Massively Parallel Neural Circuits for Stereoscopic Color Vision

The sensation of light for many animals is primarily due to two of its properties. Light intensity provides information about the brightness of the scene and the shape of objects, while its wavelength provides information about the color. Although it has long been known that the separation of color space in humans and some of the primates is mediated by three types of cone photoreceptors having peak sensitivity at different wavelengths (roughly corresponding to red, green and blue light), the exact nature of color processing and representation downstream of photoreceptors remains to be elucidated [34, 175].

The study of color vision is complicated by the fact that the early visual system employs a large number of neurons to process and represent visual stimuli. This processing involves a variety of cell types, complex wiring of neurons into canonical circuits and a large number of outputs [33, 70, 113, 131]. Furthermore, neurons in the early visual system appear to combine signals from different types of cones, thereby mixing colors on the level of individual cells so that a spike train of a single cell carries information about multiple colors present in the visual scene [34, 175]. For example, retinal ganglion cells (RGCs) exhibit opponent channel processing, whereby differences between the responses of cones such as red vs. green or blue vs. yellow or black vs. white is encoded.

How is color information represented in the activity of different types of visual neurons? Can we identify the color mixing performed by any given neuron? Can the color information be demixed once it is encoded by spiking neurons into an unlabeled set of spike trains?

In this chapter we approach these questions by proposing and studying a novel class of massively parallel neural circuits with spiking neurons that model the encoding of color visual scenes in early visual systems. For this novel class of circuits, we investigate three different but interrelated problems: encoding, decoding and functional identification. Functional identification will be investigated in Section 5.

4.1 Modeling Color Visual Stimuli

In Section 3.1.2, we modeled monochromatic visual stimuli $u = u(x, y, t), (x, y, t) \in \mathbb{D}$, as elements of an RKHS $\mathcal{H}_3^s$. Here we consider a discrete representation of spectrum, which is naturally provided by multiple types of cone photoreceptors having different peak spectral sensitivities. For example, it is well known that the trichromacy in human vision arises as a result of the visual space being sampled by three different kinds of photoreceptors at the very first stage of the visual system. Specifically, the L-, M-, and S-cones of the retina provide an initial representation of the visual space in terms of the red, green, and blue color channels, respectively. Subsequent processing within and across these color channels affords enhanced scene segmentation, visual memory, as well as perception and recognition of objects and faces [62, 64, 166].

We now extend the problem setting presented above to the space of color stimuli. Without loss of generality, we assume the traditional red, green and blue, or RGB, color representation. We assume that color visual stimuli are elements of the space of trigonometric polynomials. Each visual stimulus $\mathbf{u}$ is a vector-valued function $\mathbf{u} : \mathbb{D} \to \mathbb{R}^3$ of the form

$$\mathbf{u}(x, y, t) = [u_1(x, y, t), u_2(x, y, t), u_3(x, y, t)]^{\mathsf{T}}, \tag{4.1}$$

where each of the component functions u_1 (red channel), u_2 (green channel) and u_3 (blue channel) is a scalar-valued function in the RKHS $\mathcal{H}_3^s$. As the space we have in mind is a direct sum of three orthogonal spaces $\mathcal{H}_3^s$, we denote this color visual stimulus space as $\mathcal{H}_3^{s|3}$.

For simplicity, we assume that the bandwidth and order of each of the considered subspaces are the same. By construction, the space $\mathcal{H}_3^{s|3}$ is endowed with the inner product

$$\langle \mathbf{u}, \mathbf{w} \rangle_{\mathcal{H}_3^{s|3}} = \sum_{m=1}^{3} \langle u_m, w_m \rangle_{\mathcal{H}_3^s}. \tag{4.2}$$

RKHSs with vector-valued function elements have been studied in depth (see [21, 23] and reference within) and the reproducing kernel of $\mathcal{H}_3^{s|3}$ is given by $\mathbf{K} : \mathbb{D} \times \mathbb{D} \to \mathbf{M}(3, \mathbb{C})$, where $\mathbf{M}(3, \mathbb{C})$ is the space of 3×3 matrices (bounded linear operators on $\mathbb{R}^3$) given by

$$\mathbf{K} = \begin{bmatrix} K_1 & 0 & 0 \\ 0 & K_2 & 0 \\ 0 & 0 & K_3 \end{bmatrix}. \tag{4.3}$$

and K_m, $m = 1, 2, 3$, are reproducing kernels of $\mathcal{H}_3^s$ as in (3.4). The reproducing property of $\mathcal{H}_3^{s|3}$ is given by

$$\langle \mathbf{u}, \mathbf{K}_{x,y,t}\mathbf{v} \rangle_{\mathcal{H}_3^{s|3}} = \langle \mathbf{u}(x, y, t), \mathbf{v} \rangle_{\mathbb{R}^3}, \quad \text{for all } \mathbf{u} \in \mathcal{H}_3^{s|3} \textit{ and } \mathbf{v} \in \mathbb{R}^3, \tag{4.4}$$

where $\mathbf{K}_{x,y,t}\mathbf{v} \in \mathcal{H}_3^{s|3}$ and is defined as

$$\mathbf{K}_{x,y,t}\mathbf{v} = \mathbf{K}(x', y', t'; x, y, t)\mathbf{v}. \tag{4.5}$$

From the above, it is easy to see that for a unit vector $\mathbf{e}_m \in \mathbb{R}^3$, $m = 1, 2, 3$,

$$u_m(x, y, t) = \langle \mathbf{u}, \mathbf{K}_{x,y,t}\mathbf{e}_m \rangle_{\mathbb{R}^3}. \tag{4.6}$$

Note that the dimension of $\mathcal{H}_3^{s|3}$ is $dim(\mathcal{H}_3^{s|3}) = 3 \cdot dim(\mathcal{H}_3^s)$.

4.2 Neural Encoding Circuits for Color Vision

We now describe how the color visual stimuli of the previous section can be faithfully encoded into a multidimensional sequence of spikes by a population of spiking neurons. We employ a massively parallel neural circuit consisting of thousands of neurons, in which each neuron is a fundamentally slow device capable of producing only a limited number of spikes per unit of time. Our goal is to devise a set of conditions on the population of neurons that guarantees a faithful, or loss-free, representation of color visual stimuli in the spike domain.

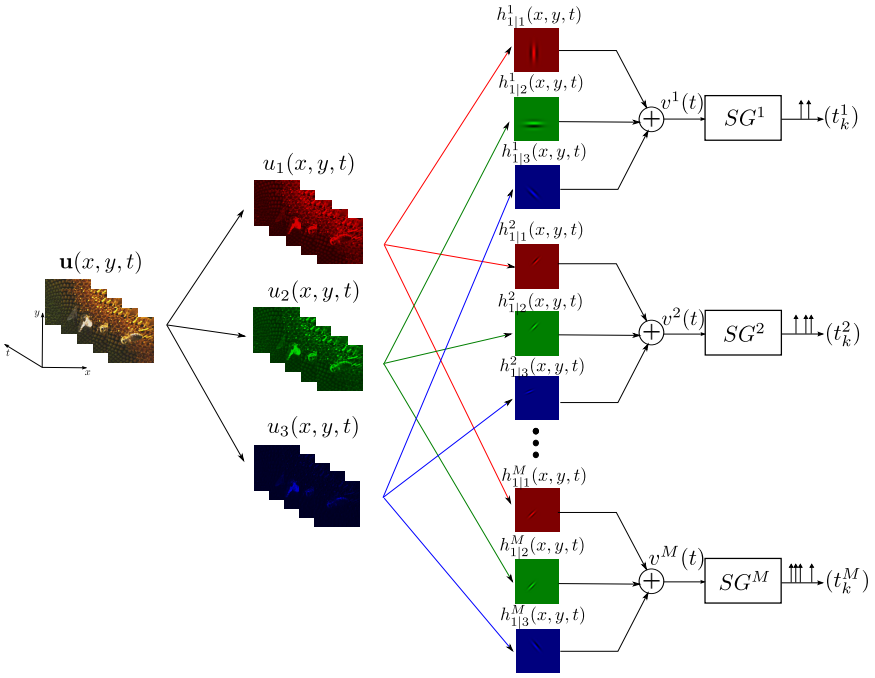


Figure 4.1: Diagram of the Encoding Neural Circuit for Color Visual Stimuli. (Modified from [118] with permission, ©2015 Elsevier)

Consider the massively parallel neural circuit shown in Figure 4.1. The color visual stimulus $\mathbf{u}$ consists of 3 components u_1, u_2, u_3, as in (4.1). These components (corresponding to the red, green, and blue channel, respectively) are assumed to be extracted by the photoreceptors and subsequently encoded by a population of M neurons. In the most general case, all neurons receive information from each photoreceptor type and multiplex (mix) and encode that information in the spike domain. Specifically, each neuron i is associated with a multi-component DSP (linear receptive field) $\mathbf{h}_1^i(x, y, t)$, where

$$\mathbf{h}_1^i(x, y, t) = \left[h_{1|1}^i(x, y, t), h_{1|2}^i(x, y, t), h_{1|3}^i(x, y, t)\right]^{\mathsf{T}}. \tag{4.7}$$

The components $h_{1|m}^i(x, y, t), m = 1, 2, 3, i = 1, \cdots, M$, are assumed to be causal in the time domain $\mathbb{D}_t$ and have a finite support in the spatial domains $\mathbb{D}_x$ and $\mathbb{D}_y$. In addition, we assume that all components of the kernel are bounded-input bounded-output (BIBO) stable. Therefore, the component filters belong to the filter kernel space $H = \mathbb{L}^1(\mathbb{D})$, where $\mathbb{L}^1(\mathbb{D})$ denotes the space of absolute integrable functions on $\mathbb{D}$.

For every neuron i, each color channel $u_m(x, y, t)$, $m = 1, \cdots, 3$, of the input signal $\mathbf{u}$ is independently filtered by the corresponding component $h_{1|m}^i(x, y, t)$ of the receptive field $\mathbf{h}_1^i(x, y, t)$, yielding a temporal signal

$$v_m^i(t) = \int_{\mathbb{D}} h_{1|m}^i(x, y, t - s) u_m(x, y, s) dx dy ds, \quad m = 1, 2, 3. \tag{4.8}$$

The outputs of the three receptive field components are then summed to provide an aggregate temporal input $v^i(t)$ to the ith neuron that amounts to

$$v^i(t) = \sum_{m=1}^{3} v_m^i(t) = \sum_{m=1}^{3} \left(\int_{\mathbb{D}} h_{1|m}^i(x, y, t - s) u_m(x, y, s) dx dy ds \right). \tag{4.9}$$

The three-dimensional color visual stimulus $\mathbf{u}(x, y, t)$ is effectively transformed into a one-dimensional signal $v^i(t)$, in which colors, spatial and temporal attributes of $\mathbf{u}$ are multiplexed. v^i is then encoded by the ith neuron into a spike train, with the sequence of spike times denoted by $(t_k^i)_{k \in \mathbb{Z}}$, where k is the spike index. The summation here can be justified by experiments in retina and V1 that have shown that the response of many neurons can be captured by a linear combination of cone signals [63, 175].

For simplicity, we assume here that each point neuron is an Integrate-and-Fire (IAF) neuron, as illustrated in Figure 4.2. However, BSGs can also be employed similarly to its use in Section 3.

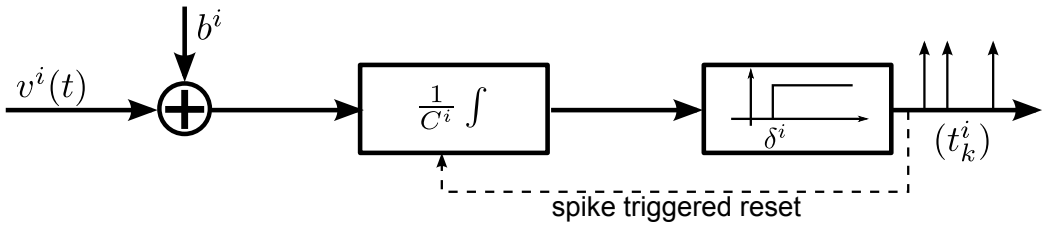


Figure 4.2: Block diagram of an (ideal) IAF neuron. The input v^i together with an additive bias b^i are passed through an integrator with integration constant C^i. A spike is generated whenever the integrator output reaches a threshold δ^i. The integrator is reset after every spike. (Modified from [118] with permission, ©2015 Elsevier)

The IAF neuron i encodes its input v^i into the sequence of spike times t_k^i

$$\int_{t_k^i}^{t_{k+1}^i} v^i(s)ds = q_k^i, \quad k \in \mathbb{Z}, \tag{4.10}$$

where $q_k^i = C^i\delta^i - b^i(t_{k+1}^i - t_k^i)$. Here C^i, δ^i and b^i are the integration constant, threshold and bias, respectively, of the ith neuron. The encoding performed by the entire neural circuit can then be expressed by the following equations

$$\int_{t_k^i}^{t_{k+1}^i} \sum_{m=1}^{3} \left(\int_{\mathbb{D}} h_{1|m}^i(x,y,t-s)u_m(x,y,s)dxdyds \right) dt = q_k^i, \quad k \in \mathbb{Z}, \tag{4.11}$$

for all $i = 1, 2, \cdots, M$. By defining linear functionals (with abuse of notation) $\mathcal{T}_k^i : \mathcal{H}_3^{s|3} \to \mathbb{R}$, $k \in \mathbb{Z}, i = 1, 2, \cdots, M$, where

$$\mathcal{T}_k^i \mathbf{u} = \int_{t_k^i}^{t_{k+1}^i} \sum_{m=1}^{3} \left(\int_{\mathbb{D}} h_{1|m}^i(x,y,t-s)u_m(x,y,s)dxdyds \right) dt, \tag{4.12}$$

equation (4.11) can be compactly rewritten as

$$\mathcal{T}_k^i \mathbf{u} = q_k^i. \tag{4.13}$$

Called the t-transform [98], equation (4.13) above describes the mapping of the analog signal $\mathbf{u}$ into a set of spikes $(t_k^i), k \in \mathbb{Z}, i = 1, 2, \cdots, M$.

By combining signals from different channels, each neuron may now carry different types of color information. For example, combining all three channels may provide luminance of the visual stimulus over a wide spectrum. Color opponency in the retina that typically takes the form of red versus green, blue versus yellow can be modeled as well.

The neural encoding circuit shown in Figure 4.1 is called the Color Video Time Encoding Machine (Color Video TEM). The Color Video TEM can be equally interpreted as a Multiple-Input Multiple-Output (MIMO) neural encoder [99], where $u_m, m = 1, 2, 3$, are seen as three separate inputs. By modeling the color video as a single element in $\mathcal{H}_3^{s|3}$, we highlight the fact that color is an intrinsic property of a natural visual stimulus.

By abuse of notation, we denote the set of indices k of spikes generated by neuron i in a finite time duration as $\mathbb{I}^i$.

Lemma 2. *The Color Video TEM projects the stimulus* $\mathbf{u}$ *onto the set of sampling functions* $\boldsymbol{\phi}_k^i = \left[\phi_{1k}^i, \phi_{2k}^i, \phi_{3k}^i\right]^\mathsf{T}$ *with* $\phi_{mk}^i = \mathcal{T}_k^i \overline{\mathbf{K}_{xyt}\mathbf{e}_m}, m = 1, 2, 3$, *and*

$$\langle \mathbf{u}, \boldsymbol{\phi}_k^i \rangle_{\mathcal{H}_3^{s|3}} = q_k^i, \quad k \in \mathbb{I}^i, \quad i = 1, 2, \cdots, M. \tag{4.14}$$

Proof: By the Riesz Representation Theorem [10], there exist functions $\boldsymbol{\phi}_k^i \in \mathcal{H}_3^{s|3}$ such that for all $\mathbf{u} \in \mathcal{H}_3^{s|3}$,

$$\mathcal{T}_k^i \mathbf{u} = \langle \mathbf{u}, \boldsymbol{\phi}_k^i \rangle_{\mathcal{H}_3^{s|3}}, \quad k \in \mathbb{I}^i, \quad i = 1, 2, \cdots, M, \tag{4.15}$$

and therefore, the encoding of the color video $\mathbf{u}$ by the TEM can be expressed as

$$\langle \mathbf{u}, \boldsymbol{\phi}_k^i \rangle_{\mathcal{H}_3^{s|3}} = q_k^i, \quad k \in \mathbb{I}^i, \quad i = 1, 2, \cdots, M.$$

The entries ϕ_{mk}^i of the sampling function $\boldsymbol{\phi}_k^i$ can be obtained by the reproducing property

$$\phi_{mk}^i(x, y, t) = \langle \boldsymbol{\phi}_k^i, \mathbf{K}_{xyt}\mathbf{e}_m \rangle = \overline{\langle \mathbf{K}_{xyt}\mathbf{e}_m, \boldsymbol{\phi}_k^i \rangle} = \mathcal{T}_k^i \overline{\mathbf{K}_{xyt}\mathbf{e}_m}, \quad m = 1, 2, 3. \tag{4.16}$$

□

In appendix C.1, we demonstrate how ϕ_{mk}^i can be efficiently computed. Thus, similar to the monochrome video encoding [98], the encoding of the color video has a simple geometrical interpretation as sampling of $\mathbf{u}$ by a set of input dependent sampling functions (coordinates)

$\boldsymbol{\phi}_k^i$, and the q_k^i, $k \in \mathbb{I}^i, i = 1, 2, \cdots, M$, are the corresponding measurements. The important difference is that the set of coordinates is not fixed. Rather, it changes at every spike time t_k^i.

4.3 Decoding Algorithms for Color Vision

4.3.1 Time Decoding Machines for Color Vision

Assuming that all receptive fields and parameters of the neurons are known, the decoding algorithm reconstructs the video $\mathbf{u}$ from the set of M spike trains $(t_k^i), k \in \mathbb{I}^i, i = 1, 2, \cdots, M$, produced by the encoding neural circuit.

Given the assumption that $\mathbf{u} \in \mathcal{H}_3^{s|3}$, and the fact that encoding of the visual stimuli consists of projections of $\mathbf{u}$ onto a set of sampling functions, we formulate the reconstruction of the encoded video as the spline interpolation problem

$$\hat{\mathbf{u}} = \underset{\mathbf{u}\in\mathcal{H}_3^{s|3},\{\mathcal{T}_k^i\mathbf{u}=q_k^i\}_{k\in\mathbb{I}^i}^{i=1,\cdots,M}}{\operatorname{argmin}} \{\|\mathbf{u}\|^2_{\mathcal{H}_3^{s|3}}\}. \tag{4.17}$$

Theorem 3. *Let the color video $\mathbf{u} \in \mathcal{H}_3^{s|3}$ be encoded by the color Video TEM with M neurons and M linearly independent receptive fields. The color video can be reconstructed as*

$$\hat{\mathbf{u}} = \sum_{i=1}^{M} \sum_{k\in\mathbb{I}^i} c_k^i \boldsymbol{\phi}_k^i, \tag{4.18}$$

where the c_k^i's are the solution to the system of linear equations

$$\boldsymbol{\Phi}\mathbf{c} = \mathbf{q}, \tag{4.19}$$

with

$$\mathbf{c} = \left[c_1^1, c_2^1, \cdots, c_{|\mathbb{I}^1|}^1, \cdots, c_1^M, c_2^M, \cdots, c_{|\mathbb{I}^M|}^M\right]^\mathsf{T},$$
$$\mathbf{q} = \left[q_1^1, q_2^1, \cdots, q_{|\mathbb{I}^1|}^1, \cdots, q_1^M, q_2^M, \cdots, q_{|\mathbb{I}^M|}^M\right]^\mathsf{T},$$

and $\boldsymbol{\Phi}$ is the block matrix

$$\boldsymbol{\Phi} = \left[\boldsymbol{\Phi}^{ij}\right], \tag{4.20}$$

where $i, j = 1, 2, \cdots, M$, and the block entries are given by (see also Section C.1)

$$\left[\mathbf{\Phi}^{ij}\right]_{kl} = \langle \boldsymbol{\phi}_k^i, \boldsymbol{\phi}_l^j \rangle_{\mathcal{H}_3^{s|3}}, \text{ for all } k \in \mathbb{I}^i, l \in \mathbb{I}^j, i, j = 1, 2, \cdots, M.$$

A necessary condition for perfect recovery of any arbitrary $\mathbf{u} \in \mathcal{H}_3^{s|3}$ is that the set of sampling functions $\boldsymbol{\phi} = (\boldsymbol{\phi}_k^i), k \in \mathbb{I}^i, i = 1, 2, \cdots, M$, span $\mathcal{H}_3^{s|3}$. This requires to have the number of neurons $M \geq 3 \cdot dim_{xy}(\mathcal{H}_3^s)$ and a total of at least $\dim(\mathcal{H}_3^{s|3}) + M$ spikes.

Proof: See appendix B.3 for a complete proof. □

Theorem 3 shows that color visual stimuli can be reconstructed by a linear combination of sampling functions. Moreover, color visual stimuli can be perfectly reconstructed only if the encoder generates enough spikes such that the sampling functions fully explore the input space.

Remark 3. *The solution of the optimization problem in Theorem 3 can easily be extended to an infinite dimensional space with $\hat{\mathbf{u}}$ reconstructed in the subspace generated by the span of the set of sampling functions $(\boldsymbol{\phi}_k^i)$, $k \in \mathbb{I}^i$, $i = 1, 2, \cdots, M$. Moreover, (4.19) can be solved by the recurrent neural network algorithm describe in Section 3.4.1.*

The overall decoding procedure is summarized in the diagram shown in Figure 4.3.

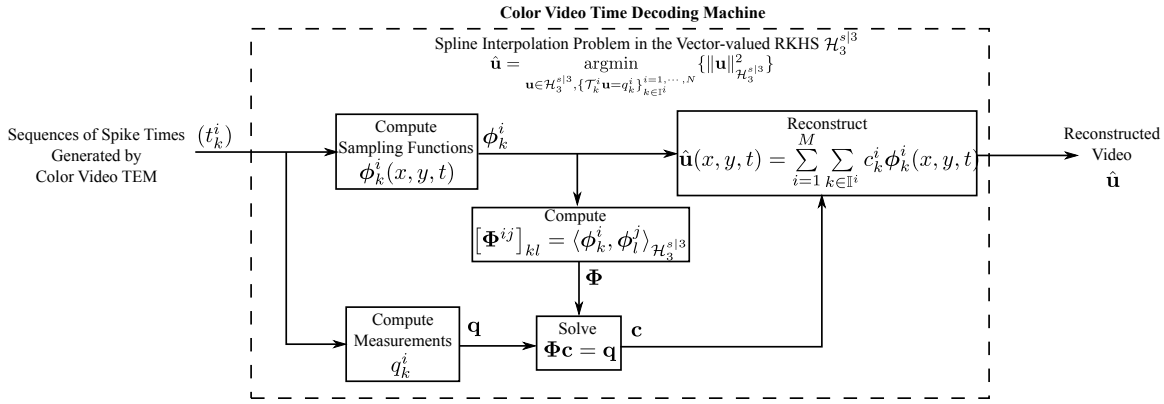


Figure 4.3: Color Video TDM diagram. (Modified from [118] with permission, ©2015 Elsevier)

4.3.2 Evaluating Massively Parallel TDM Algorithms for Color Vision

In this section we describe an example of encoding and decoding of a color video sequence. The stimulus is a 10 [s], 160[px] $\times$ 90[px] natural color video. The video is encoded by a neural circuit consisting of $30,000$ color-component receptive fields in cascade with $30,000$ IAF neurons. The detailed construction of the encoding circuit is given in appendix C.2.

In reconstruction, a spatio-temporal stitching algorithm similar to the one in [108] is deployed. The entire screen is divided into 4 overlapping regions of size 56[px] $\times$ 90[px] and time is cut into 150 [ms] slices. The stitching volume then becomes 56[px] $\times$ 90[px] $\times$ 0.15[s]. We picked the orders of the space to be $L_x = 24$, $L_y = 36$, $L_t = 8$, $\Omega_x = \Omega_y = 0.375 \cdot 2\pi$ and $\Omega_t = 10 \cdot 2\pi$ so that the overall period of the space is larger than each of the volumes. We did this in order to embed a typically non-periodic natural stimulus into a periodic space.

Remark 4. *Note that natural stimuli exhibit an "$1/f$ spectrum". Their Fourier coefficients are typically significant only in a certain spectral neighborhood around the origin. For simulation purposes, we restricted the set of spatial basis functions (l_x, l_y) to lie inside the elliptical set $\{(l_x, l_y) \,|\, l_x^2/L_x^2 + l_y^2/L_y^2 \leq 1\}$. Since the coefficients of all other basis functions in $\mathcal{H}$ are set to zero this restriction let us work instead in a subspace of $\mathcal{H}$ with an elliptical bandwidth profile. The spatial dimension $\dim_{xy}(\mathcal{H}_3^s)$ is thereby approximated by the cardinality of the above subspace. Throughout this chapter, if not stated otherwise, the cardinality of this subspace will provide the order of $\mathcal{H}_3^s$.*

The total number of spikes produced by encoding a 10-second long video was $9,001,700$. Each volume is typically reconstructed from about $90,000$ measurements. It took about 2 days to reconstruct the video on 9 GPUs. We show the original color video and the reconstructed video in the video in Figure 4.4. The original video is shown on the left, which shows a bee on a sunflower. The reconstruction is shown in the middle and the error on the right. The error can be seen to be fairly small (zero error is shown in gray). The signal-to-noise ratio (SNR) of the reconstruction is 30.71 [dB]. The structural similarity (SSIM) index [190] of the reconstruction is 0.988. In addition, each color component can be individually accessed. Demixed videos from all three channels are shown in the video in Figure 4.5. Since the original video is a natural video and is not strictly in the RKHS, the

video is not perfectly reconstructed. However, it is still decoded with very high quality.

Figure 4.4: A snapshot of the original and the reconstructed color video. From left to right are respectively, the original video, reconstructed color video and the error. (Modified from [118] with permission, ©2015 Elsevier)

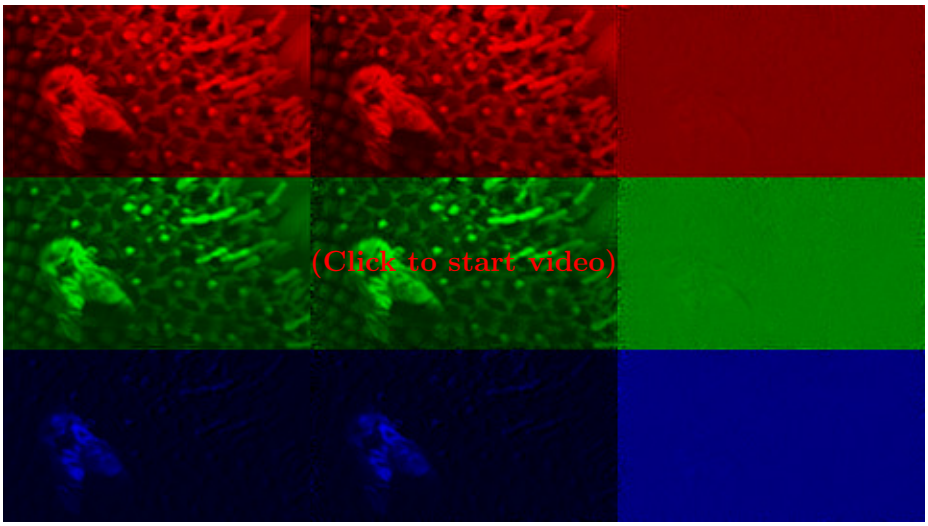


Figure 4.5: A snapshot of the original and the reconstructed videos of each color channel. From left to right are respectively, the original video, reconstructed color video and the error. Each color channel is a gray-scale image that is pseudo-colored to indicate the respective channels. (Modified from [118] with permission, ©2015 Elsevier)

A limitation in modeling stimuli using the representation in (3.1) is that stimuli must be periodic. However, this is rarely true for natural visual stimuli. The imposed periodicity can be mitigated by using a larger period in reconstruction. We note that the periods S_x, S_y, S_t defined in the reconstruction space are larger than the size and duration of the stitching volume. By embedding the stitching volume in a larger periodic space, the reconstruction no longer has to be periodic within the stitching volume. This makes reconstruction of

natural stimuli possible and the choice of space flexible. It may seem at first that the dimension of the space that the reconstruction is embedded into is increased. However, this does not necessarily mean that the number of spikes required to reconstruct the stimulus has to increase. This is due to the fact that the sampling functions only need to span a subspace that the stitching volume is associated with. For example, in the reconstruction above, the space of choice is of dimension $137,751$ ($dim_{xy}(\mathcal{H}_3^s) = 2,701$, $dim_t(\mathcal{H}_3^s) = 17$). Already some $90,000$ measurements yield a high quality reconstruction of the visual stimuli. We note that the temporal period is $S_t = 0.8s$, while the duration of the stitching volume is only 0.15s. As long as the spikes are dense enough within the 0.15s time interval of interest, the stimulus can be reconstructed even if the number of spikes is lower than $137,751$. However, the sampling functions still have to span a minimal subspace that encloses the 0.15 time duration.

4.4 Massively Parallel Neural Circuit for Stereoscopic Color Vision

4.4.1 Massively Parallel Neural Circuits for Stereoscopic Video

Stereoscopic videos are two different streams of video that are projected onto the left and right eyes. Typically, the two video streams represent views of the same visual scene taken from slightly different angles. They arise naturally in the early visual system of vertebrates where binocular vision dominates. By combining multiple views of the visual scene, binocular vision makes it possible to extract the depth information about the visual scene.

A massively parallel neural circuit for encoding monochrome (grayscale) stereoscopic video is shown in Figure 4.6. The input videos, denoted by (abuse of notation),

$$\mathbf{u}(x, y, t) = [u_1(x, y, t), u_2(x, y, t)]^{\mathsf{T}},$$

may come from a single visual scene but are sensed by two eyes, where u_1 denotes the monochrome video sensed by the left eye and u_2 denotes that sensed by the right eye. In the visual cortex, the information from both eyes is combined in some of the neurons [155].

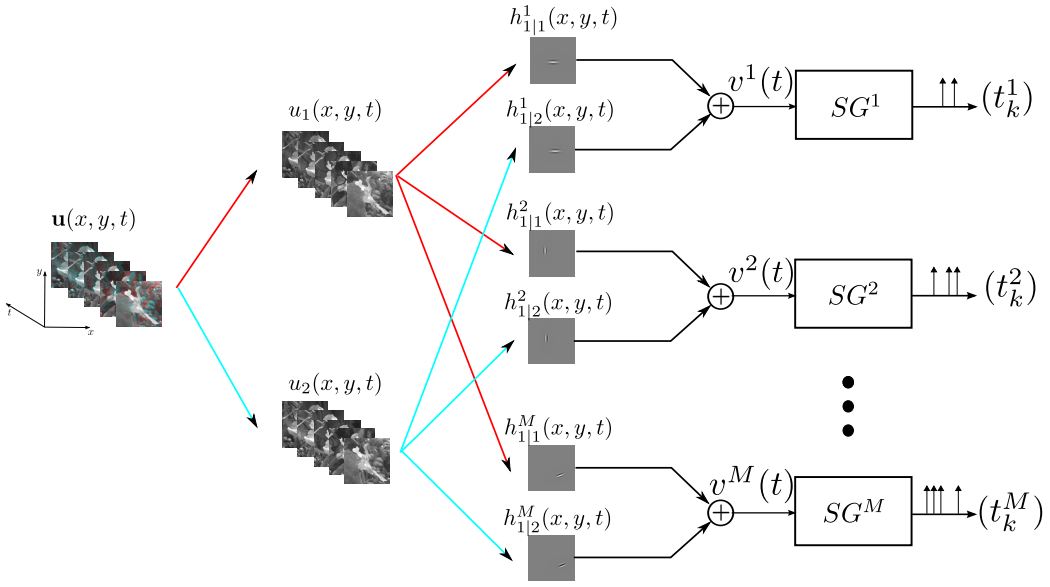


Figure 4.6: Diagram of massively parallel neural circuit for encoding stereoscopic video. (Modified from [118] with permission, ©2015 Elsevier)

This is modeled by the multi-component DSPs $\mathbf{h}^i(x,y,t)$, where, by abuse of notation,

$$\mathbf{h}^i_1(x,y,t) = \left[h^i_{1|1}(x,y,t), h^i_{1|2}(x,y,t)\right]^{\mathsf{T}}. \tag{4.21}$$

Again, each component $h^i_{1|m}(x,y,t), m = 1,2, i = 1,\cdots,M$, is assumed to be causal with finite support, and BIBO stable. Each component receptive field performs a linear filtering operation on its corresponding input video before the outcomes are summed and fed into an IAF neuron [57, 205]. The above neural encoding circuit forms a Stereoscopic Video TEM.

We provide an example here demonstrating the encoding of stereoscopic videos and their reconstruction. The stereoscopic video has a view of 192 [px] × 108 [px] in each component and was shot by two cameras calibrated to match binocular vision and provide a 3D visual perception [80]. Parameters of the space are $L_x = 72$, $L_y = 40$, $\Omega_x = 0.75\pi$, $\Omega_y = 0.74\pi$, $L_t = 8$ and $\Omega_t = 20\pi$. The reconstruction is shown in the video in Figure 4.7. SNR of the reconstruction is 35.77dB, SSIM index is 0.980. The reconstructions of separate eye channels are shown in the video in Figure 4.8.

Figure 4.7: The original stereo video and the reconstructed stereo video. From left to right are respectively, the original video, reconstructed stereo video and the error. The 3D effects can be visualized by wearing red-cyan 3D glasses. (Modified from [118] with permission, ©2015 Elsevier)

Figure 4.8: The original stereo video and the reconstructed stereo video in separate channels. The left eye channel is shown in the top row and the right eye channel in the bottom row. From left to right are respectively, the original video, reconstructed video and the error. (Modified from [118] with permission, ©2015 Elsevier)

4.4.2 Massively Parallel Neural Circuits for Stereoscopic Color Video

The massively parallel neural circuits for color video and stereoscopic video can be combined to work with stereoscopic color video. The RKHS of interest then becomes $\mathcal{H}_3^{s|6}$. Figure 4.9 depicts a block diagram of the massively parallel neural circuit for encoding stereoscopic color video. Neurons in the circuit can now encode information in all the color channels of both eyes.

The encoding and decoding based on this circuit can be formulated similarly as described in Section 4.2 and 4.4.1.

We show here a video in the video in Figure 4.10 for demonstration of the decoding of stereo color video. SNR of the reconstruction is 27.37dB, SSIM index is 0.960. The reconstruction of individual channels are shown in the video in Figure 4.11.

4.5 Summary

In this chapter, we presented TEMs and derived TDMs for color and stereoscopic visual stimuli. A common feature of encoding of all these stimuli is the use of multiple sensors to extract and to subsequently combine information from these sensors. Color visual scenes were decomposed into three color channels. Neurons then sampled, compared or composed information contained in the output of various color channels and multiplexed that information in the time domain using spikes. For stereoscopic vision, the visual scene was separately sensed by two horizontally displaced observers (eyes) and fed into a population of neurons. The receptive fields of each neuron can individually process and compose the information received from both eyes.

Natural scenes are highly complex with variations in intensity, wavelength and geometry. It is interesting to note that in order to perceive the complexity of the visual world, the visual system seems to mix different types of information. The TEMs we formulated here for stereoscopic color vision are instances of such mixing.

Mixed signals encoding may be important in a number of ways. First, each of the color channels represents an aspect of a visual scene. Information can be highly redundant across multiple channels. For example, all RGB channels carry information about the same objects

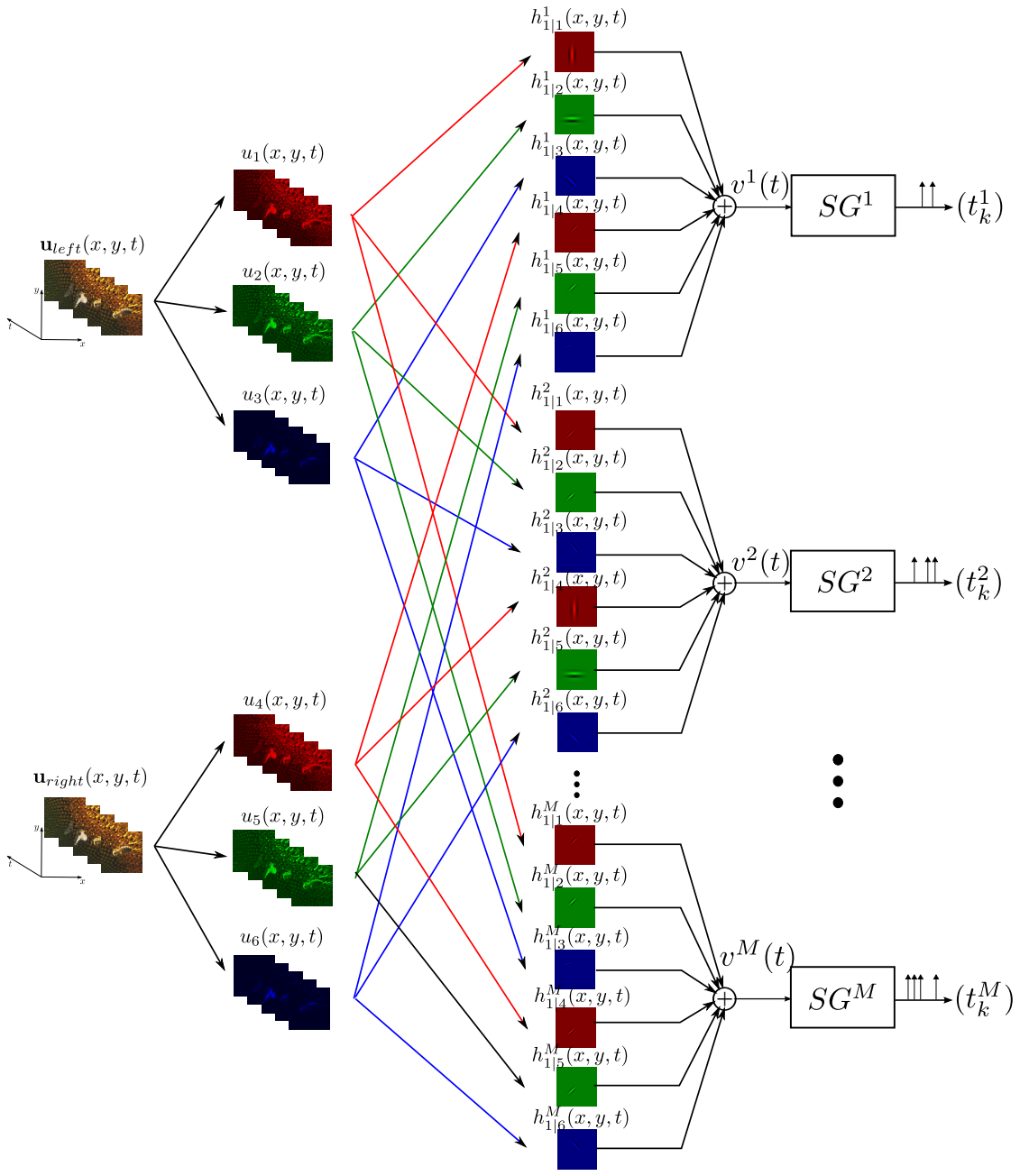


Figure 4.9: Diagram of the massively parallel neural circuit for encoding stereoscopic color video. (Modified from [118] with permission, ©2015 Elsevier)

Figure 4.10: The original stereoscopic color video and the reconstructed stereoscopic color video. From left to right are respectively, the original video, reconstructed color video and the error. The 3D effects can be visualized by wearing red-cyan 3D glasses. (Modified from [118] with permission, ©2015 Elsevier)

in a visual scene. The shapes of these objects are shared in all channels. A change in color intensity is more likely to happen at the boundary between two objects and this change is propagated across color channels. Combining information from multiple channels may enhance the encoding efficiency and provide a simpler decoding algorithm. The YUV or YCbCr video formats, for example, have long been used in digital video technology where some of the channels can be subsampled while, at the same time, keeping a similar perceptual quality level. We did not explore here such a method of redundancy reduction. Rather, we provided a framework for representing multiple channels of information, for recovering the scene and for identifying channel parameters, such that these facilitate redundancy reduction.

Second, the mixing of cone signals can be utilized as coordinate transformations in the space/time color space. Such transformations may be useful in object recognition or in the separation of color and contrast information.

Third, mixing multiple channel output signals allows multiple information patterns to be represented together and therefore enables readout of different aspects of signals anywhere in the system. In other words, it provides broadcast information to multiple "receivers". This makes it possible for higher level neural systems to extract information from a common pool of spikes.

Figure 4.11: The original stereo color video and the reconstructed in separate channels. The first three rows are the color channels in the left eye and the last three rows are the color channels in the right eye. From left to right are respectively, the original video, reconstructed video and the error. (Modified from [118] with permission, ©2015 Elsevier)

Chapter 5

Functional Identification of Massively Parallel Neural Circuits and its Evaluation

Functional identification aims at inferring the transformation performed by a neural circuit or system. In the setting of functional identification, the inputs are known and the outputs are observable, while the functionality of the neural circuit is unknown. Functional identification is critical in understanding the information processing taking place in biological systems and is operationally significant to decipher what a neural circuit does.

Functional identification of a neural circuit is formulated here in the setting of CIMs [102]. CIMs are algorithms that identify neural circuit parameters (e.g., receptive fields and parameters of spike generators) directly from spike times generated in response to a collection of test stimuli. Some of the key advantages of parameter identification in the setting of CIMs are (i) clear lower bounds on the number of test stimuli and spikes required for identification can be specified and (ii) both synthetic and naturalistic stimuli can be used [103, 104].

In addition, we formulate CIMs for functionally identifying linear DSPs that carry visual information from multiple color channels. This allows us to identify the spatio-temporal

This chapter appears, in part, in [118] ©2015 Elsevier.

structure of the DSPs in each of the base color channels.

Moreover, we extend the duality between TDMs and CIMs [102] and thereby obtain a general theory of neural representation of color visual information. We provide examples to demonstrate that a massively parallel color video encoding neural circuit of unknown parameters can be identified with arbitrary precision and its spike trains can be used to reconstruct the encoded stimuli.

Given the extended duality between TEMs and CIMs, we argue that evaluation of the functional identification methodology can be effectively and intuitively performed in the stimulus space. In this space the reconstructed signals using spike trains generated by an identified neural circuit can be compared to the original stimulus.

We also provide rigorous methods for identifying circuits in the early visual system. In Section 5.2 we present an unified framework for quantitatively evaluating identification algorithms of massively parallel neural circuits in the stimulus space and demonstrate the use of decoding algorithms as a means of model verification.

5.1 Identification of Neural Encoding Circuits for Color Vision

5.1.1 Channel Identification Machines for Color Vision

The color video encoded by the Color Video TEM can be reconstructed, given the spike times produced by a population of neurons and the parameters of each of the neurons. However, in many circumstances, the parameters of the neurons are not available a priori and need to be identified. In this scenario, the neurons are typically presented with one or more input test stimuli and their response, or output, is recorded so that neuron parameters can be identified using the input/output data. It can be shown that identification problems of this kind are mathematically dual to the decoding problem discussed in the previous section. Specifically, it can be shown that information about both the receptive fields and the spike generation mechanism can be faithfully encoded in the spike train of a neuron. Spike times are viewed as signatures of the entire system, and under appropriate conditions, these signatures can be used to identify both the receptive fields and the parameters of point neurons. The key experimental and theoretical insight is that the totality of spikes produced by a single

neuron in M experimental trials can be treated as a single multidimensional spike train of a population of M neurons encoding fixed attributes of the neural circuit. Furthermore, it can be proven that only a projection of the neural circuit parameters onto the input stimulus space can be identified. The projection is determined by the particular choice of stimuli used during experiments and under natural conditions it converges to the underlying parameters of the circuit [101, 102, 103].

In this section we demonstrate that the ideas developed in [101], [102] and [103] can be extended to massively parallel neural circuits that process (color) visual stimuli. For clarity, we consider identification of receptive fields only. Identification of spike generation parameters and/or connectivity between neurons can be handled similarly to what has been previously described [103].

For presentation purposes, we consider the identification of a single receptive field associated with only one neuron, since identification of multiple receptive fields for a population of neurons can be performed in a serial fashion. We therefore drop the superscript i in $h^i_{1|m}$ throughout this section and denote the m-th kernel component by $h_{1|m}$. Moreover, we introduce the natural notion of performing multiple experimental trials and use the same superscript i to index stimuli $\mathbf{u}^i$ on different trials $i = 1, \cdots, M$. In what follows, the neural circuit referred to as the Color Video TEM consists of a color receptive field $\mathbf{h}_1 = [h_{1|1}, h_{1|2}, h_{1|3}]^\mathsf{T}$ in cascade with a single IAF neuron.

Definition 6. *A signal $\mathbf{u}^i$, at the input to a Color Video TEM together with the resulting output $\mathbb{T}^i = (t^i_k)_{k\in\mathbb{Z}}$ is called an input/output (I/O) pair and is denoted by $(\mathbf{u}^i, \mathbb{T}^i)$.*

Definition 7. *The operator $\mathcal{P}^s_1 : H^3 \to \mathcal{H}^{s|3}_3$ with elements $(\mathcal{P}^s_1\mathbf{h}_1)_m$, $m = 1, 2, 3$, given by*

$$(\mathcal{P}^s_1\mathbf{h}_1)_m(x, y, t) = \int_{\mathbb{D}} h_{1|m}(x', y', t')K_m(x, y, t; x', y', t')dx'dy'dt',$$

is called the projection operator.

Consider a single neuron receiving a stimulus $\mathbf{u}^i \in \mathcal{H}^{s|3}_3, i = 1, 2, \cdots, M$. The aggregate output $v^i = v^i(t)$, $t \in \mathbb{D}_t$, of the receptive field $\mathbf{h}_1$ produced in response to the stimulus $\mathbf{u}^i$

during the trial i is given by

$$v^i(t) = \sum_{m=1}^{3} \int_{\mathbb{D}} h_{1|m}(x, y, t-s) u_m^i(x, y, s) dxdyds, \tag{5.1}$$

where each signal u_m^i is an element of the space $\mathcal{H}_3^s$.

Let $\mathcal{L}_k^i : \mathcal{H}_3^{s|3} \to \mathbb{R}$ be the linear functionals given by

$$\mathcal{L}_k^i\left[\mathcal{P}_1^s \mathbf{h}_1\right] = \int_{t_k^i}^{t_{k+1}^i} \sum_{m=1}^{3} \left(\int_{\mathbb{D}} u_m^i(x, y, t-s) (\mathcal{P}_1^s \mathbf{h}_1)_m(x, y, s) dxdyds \right) dt, \tag{5.2}$$

for all $k \in \mathbb{I}^i, i = 1, \cdots, M$. Then, the Color Video TEM is described by the set of equations (see also Appendix B.4):

$$\mathcal{L}_k^i\left[\mathcal{P}_1^s \mathbf{h}_1\right] = q_k^i, \quad k \in \mathbb{I}^i, \quad i = 1, \cdots, M. \tag{5.3}$$

We note that because $\mathcal{L}_k^i$ is linear and bounded, (5.3) can be expressed in inner product form as

$$\langle \mathcal{P}_1^s \mathbf{h}_1, \boldsymbol{\psi}_k^i \rangle = q_k^i, \tag{5.4}$$

where $\boldsymbol{\psi}_k^i(x, y, t) = \left[\psi_{1k}^i(x, y, t), \psi_{2k}^i(x, y, t), \psi_{3k}^i(x, y, t)\right]^{\mathsf{T}}$ and

$$\psi_{mk}^i(x, y, t) = \mathcal{L}_k^i\left[\overline{\mathbf{Ke}_m}\right], \quad m = 1, 2, 3. \tag{5.5}$$

In contrast to equation (4.14) each inter-spike interval $[t_k^i, t_{k+1}^i)$ produced by the IAF neuron on experimental trial i is now treated as a quantal measurement q_k^i of the sum of the components of the receptive field $\mathbf{h}_1$, and not the stimulus $\mathbf{u}^i$. When considered together, equations (5.3) and (4.13) demonstrate that the identification problem can be converted into a neural encoding problem similar to the one discussed in the previous section. Note, however, that in (4.13) i denotes the neuron number whereas i in (5.3) denotes the trial number. This concept is further illustrated in Figure 5.1.

We note that the spike trains produced by a Color Video TEM in response to test stimuli $\mathbf{u}^i$, $i = 1, \cdots, M$, carry only partial information about the underlying receptive field $\mathbf{h}_1$. Intuitively, the information content is determined by how well the test stimuli explore the system. More formally, given test stimuli $\mathbf{u}^i \in \mathcal{H}_3^{s|3}$, $i = 1, \cdots, M$, the original receptive field $\mathbf{h}_1$ is projected onto the space $\mathcal{H}_3^{s|3}$ and only that projection $\mathcal{P}_1^s \mathbf{h}_1$ is encoded in the

neural circuit output. It follows from (5.4) that we should be able to identify the projection $\mathcal{P}_1^s \mathbf{h}_1$ from measurements q_k^i, $k \in \mathbb{I}^i$, $i = 1, \cdots, M$.

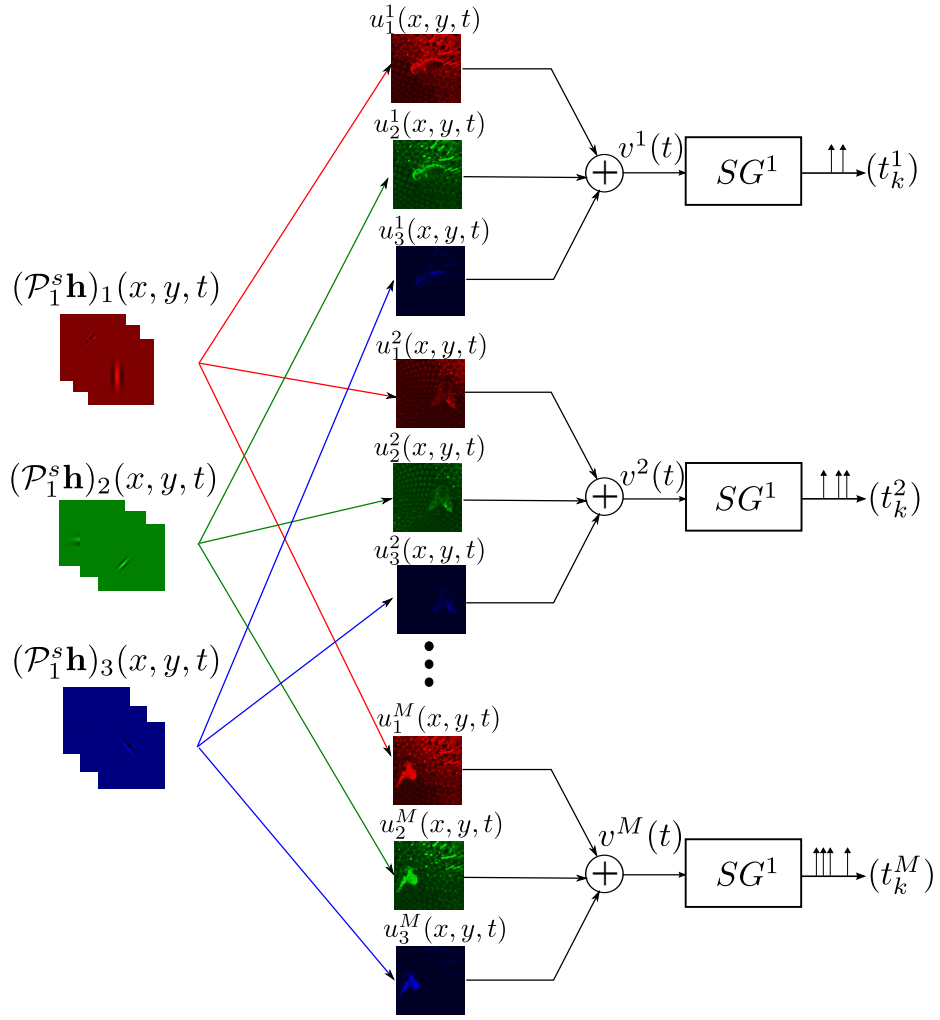


Figure 5.1: Block diagram of functional identification with multiple trials of controlled visual stimuli. Note that the same neuron is stimulated with M visual stimuli. (Modified from [118] with permission, ©2015 Elsevier)

We now provide an algorithm, called the Color Video Channel Identification Machine (Color Video CIM), to functionally identify a neural circuit processing color video stimuli. As discussed in the previous section, this algorithm can be considered to be the dual of the decoding algorithm of Theorem 3.

Theorem 4. *Let* $\{\mathbf{u}^i|\mathbf{u}^i \in \mathcal{H}_3^{s|3}\}_{i=1}^M$ *be a collection of* M *linearly independent stimuli at the input to a Color Video TEM with a receptive field* $\mathbf{h}_1$*. The projection* $\mathcal{P}_1^s\mathbf{h}_1$ *of the receptive field* $\mathbf{h}_1$*, can be identified from a collection of I/O pairs* $\{\mathbf{u}^i, \mathbb{T}^i\}_{i=1}^M$ *as a solution to the spline interpolation problem*

$$\widehat{\mathcal{P}_1^s\mathbf{h}_1} = \underset{\mathcal{P}_1^s\mathbf{h}_1 \in \mathcal{H}_3^{s|3}, \{\mathcal{L}_k^i\mathbf{h}_1 = q_k^i\}_{k\in\mathbb{I}^i}^{i=1,\cdots,M}}{\operatorname{argmin}} \{\|\mathcal{P}_1^s\mathbf{h}_1\|^2_{\mathcal{H}_3^{s|3}}\}. \tag{5.6}$$

The solution is

$$\widehat{\mathcal{P}_1^s\mathbf{h}_1} = \sum_{i=1}^{M}\sum_{k\in\mathbb{I}^i} c_k^i\boldsymbol{\psi}_k^i,$$

where the c_k^i*'s satisfy the system of linear equations*

$$\boldsymbol{\Psi}\mathbf{c} = \mathbf{q}, \tag{5.7}$$

$\mathbf{c} = \left[c_1^1, c_2^1, \cdots, c_{|\mathbb{I}^1|}^1, \cdots, c_1^M, c_2^M, \cdots, c_{|\mathbb{I}^M|}^M\right]^\mathsf{T}$, $\mathbf{q} = \left[q_1^1, q_2^1, \cdots, q_{|\mathbb{I}^1|}^1, \cdots, q_1^M, q_2^M, \cdots, q_{|\mathbb{I}^M|}^M\right]^\mathsf{T}$, *and* $\boldsymbol{\Psi}$ *is the block matrix*

$$\boldsymbol{\Psi} = \left[\boldsymbol{\Psi}^{ij}\right], \tag{5.8}$$

where $i, j = 1, 2, \cdots, M$*, and the block entries are given by (see also appendix C.1)*

$$\left[\boldsymbol{\Psi}^{ij}\right]_{kl} = \langle\boldsymbol{\psi}_k^i, \boldsymbol{\psi}_l^j\rangle_{\mathcal{H}_3^{s|3}}, \text{ for all } k \in \mathbb{I}^i, l \in \mathbb{I}^j, i, j = 1, 2, \cdots, M.$$

A necessary condition for perfect identification of $\mathcal{P}_1^s\mathbf{h}_1 \in \mathcal{H}_3^{s|3}$ *is that the set of sampling functions* $\boldsymbol{\psi} = (\boldsymbol{\psi}_k^i), k \in \mathbb{I}^i, i = 1, 2, \cdots, M$*, span* $\mathcal{H}_3^{s|3}$*. This requires to have the number of trials* $M \geq 3 \cdot dim_{xy}(\mathcal{H}_3)$ *and a total of at least* $\dim(\mathcal{H}_3^{s|3}) + M$ *spikes. Equivalently, if the neuron produces* ν *spikes on each trial* $i = 1, \cdots, M$*, of duration* S_t*, then the number of trials*

$$M \geq \begin{cases} \left\lceil \dfrac{3 \cdot dim(\mathcal{H}_3^s)}{\nu - 1} \right\rceil, & \nu < 2L_t + 2 \\ \\ 3 \cdot dim_{xy}(\mathcal{H}_3^s), & \nu \geq 2L_t + 2. \end{cases}$$

Proof: The proof is along the lines of the one for Theorem 3.

Remark 5. *Note that only the projection* $\mathcal{P}_1^s\mathbf{h}_1$ *of* $\mathbf{h}_1$ *onto* $\mathcal{H}_3^{s|3}$ *can be identified. In addition, notice the similarity of the identification and the decoding algorithms. This is a direct result of the duality of the functional identification and decoding problems.*

Theorem 4 provides, in addition to an algorithm, a lower bound on the number of video clips and the number of spikes required to perfectly identify the projection of receptive fields.

5.1.2 Evaluating Massively Parallel CIM Algorithms for Color Vision

In this section, we show an example of functional identification of a single non-separable spatio-temporal receptive field. We first demonstrate how one can use both natural video and artificially generated bandlimited noise to identify the receptive fields, and illustrate bounds for the number of video clips and number of spikes for perfect identification. Then, we perform a full-scale identification of the neural circuit described in Section 4.3.2 by using a long sequence of continuous natural stimuli instead of short video clip segments.

In the first example, the neuron to be identified has a receptive field that resembles that of a Red-On-Green-Off (R+G-) midget cell in the primate retina [9]. The red and green components of the receptive field are modeled as space-time separable functions. They are Gaussian functions spatially, and resemble bandpass linear filters temporally. The blue component will also be identified, although it is set to zero. The temporal span of the filter is 150 [ms] and spatially it is confined to a 32 [px] $\times$ 32 [px] screen size. The receptive field is shown in the video in Figure 5.2.

To identify the receptive field, we consider $S_t = 300$ms, $S_x = S_y = 32$. We chose $L_x = L_y = 6$, $L_t = 12$, $\Omega_x = \Omega_y = 0.25\pi$ and $\Omega_t = 80\pi$. The total dimension of the space is $dim_{xy}(\mathcal{H}_3^s) \times (2L_t + 1) \times 3 = 8,475$, where $\dim_{xy}(\mathcal{H}_3^s) = 113$, since we restricted the spatial basis functions to the set$\{(l_x, l_y) \,|\, l_x^2/L_x^2 + l_y^2/L_y^2 \leq 1\}$ (see also Remark 4). The projection of the receptive field $\mathcal{P}_1^s \mathbf{h}_1$, shown also in the video in Figure 5.2 is close to the underlying receptive field $\mathbf{h}_1$ itself. The Signal-to-Noise ratio (SNR) of the projection of the original filter onto the stimulus space is 23.12 [dB] (the SNR is computed only for the red and green components). In what follows, we will only depict the SNR of the identified receptive fields and compare with the projection of the receptive field.

To identify the receptive field, we generate M video clips in the Hilbert space of interest by randomly picking coefficients for each basis. Note that by randomly picking coefficients we can easily end up with complex valued signals. We further ensure that the visual stimuli are real valued.

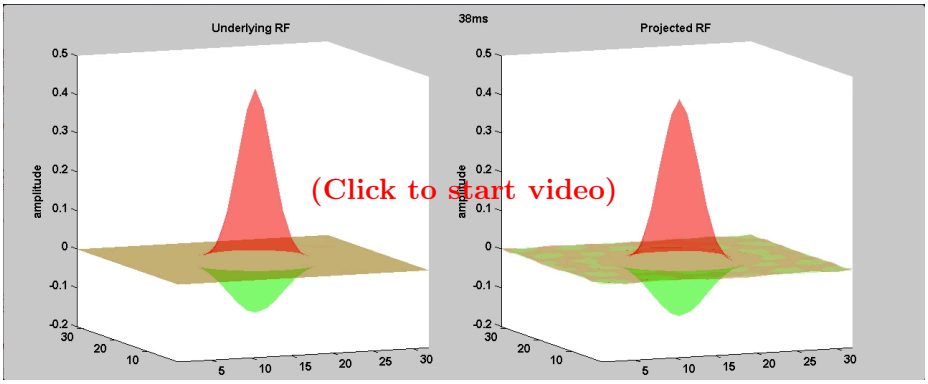


Figure 5.2: An example of a color-component spatio-temporal receptive field (150ms duration). (Left) The receptive field to be identified. (Right) The projection of the receptive field. Parameters of the elements of the space $\mathcal{H}_3^s$ are: $S_t = 300$ms, $S_x = S_y = 32$, $L_x = L_y = 6$, $L_t = 12$, $\Omega_x = \Omega_y = 0.25\pi$ and $\Omega_t = 80\pi$. The red component of the receptive field provides the excitatory center (red surface) while the green component provides the inhibitory surround (green surface). The subtle difference between the original and the projected receptive field (23.12 [dB] SNR) indicates that the chosen input space is effective in exploring the receptive field. (Modified from [118] with permission, ©2015 Elsevier)

To illustrate the identification quality for different number of spikes and number of video clips, we modified the parameters of the IAF neuron while keeping the underlying receptive field the same. Note that the modification of parameters of the IAF neuron may not be biologically plausible; it is used here to better illustrate the bounds on the number of measurements.

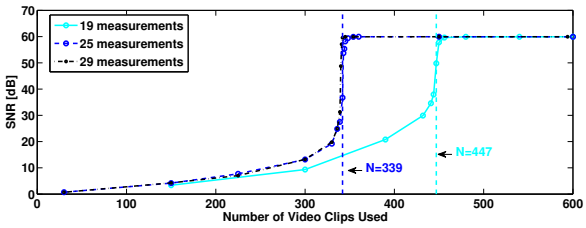
First, we vary the number of video clips M while using the same number of spikes generated by each video clip. The SNR is shown is Figure 5.3(a). All three curves follow a general trend: the SNR increases as more video clips are used until it saturates at around 60 [dB], which indicates perfect identification (up to machine precision). Comparing the three curves, we see that if the neuron produces more than 25 measurements, or 26 spikes, per each video clip (blue and black curves), the SNR saturates at roughly $M = 3 \times \dim_{xy}(\mathcal{H}_3^s) = 3 \times 113 = 339$ video clips, as stated in Theorem 4. This corresponds to the lower bound on the number of stimuli needed in order to identify the receptive field. If the neuron generates

fewer than 25 measurements, more stimuli are needed. For example, if 19 measurements, or $\nu = 20$ spikes, are produced in response to each stimulus, then a minimum of $\lceil 3 \times \dim_{xy}(\mathcal{H}_3^s) \times \dim_t(\mathcal{H}_3^s)/(\nu - 1) \rceil = \lceil 3 \times 113 \times 25/19 \rceil = 447$ video clips is needed.

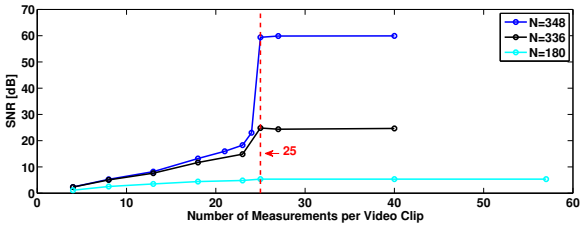
In Figure 5.3(b), we fix the number of video clips while varying the number of measurements of each video clip. Note that as the number of measurements per each video clip increases, so does the identification quality. However, the SNR cannot be further improved after the number of spikes for each video clip reaches $(2L_t + 1) + 1 = 26$ spikes, or 25 measurements. This is due to the fact that the freedom of the space in the temporal dimension is $(2L_t + 1)$. As shown in Theorem 4, once each neuron fires $(2L_t + 1) + 1 = 26$ spikes, it does not produce additionally informative measurements. Therefore, the identification quality cannot be improved further. Furthermore, the cyan and black curves demonstrate that even if the total number of measurements is larger than the dimension of the space, perfect identification may not be achieved if the lower bound $M = 3 \cdot \dim_{xy}(\mathcal{H}_3^s) = 3 \times 113 = 339$ is not met. For example, if the neuron generates 40 measurements per video clip, then both 180 and 336 video clips will result in the total number of spikes that is greater than the required number of $3 \times \dim(\mathcal{H}_3^s) = 8,475$ spikes.

In Figure 5.3(c), we vary the number of video clips while the total number of measurement remains fixed at around $9,000$. Identification quality is high when $M \geq 339$, illustrating the lower bound of videos/experiments needed in order to identify the receptive field with arbitrary precision. However, the identification quality saturates at $M = 342$, slightly shifted away from $M = 339$. This shift is mainly due to numerical errors and the random choice of video clips that does not guarantee linear independence among the sampling functions.

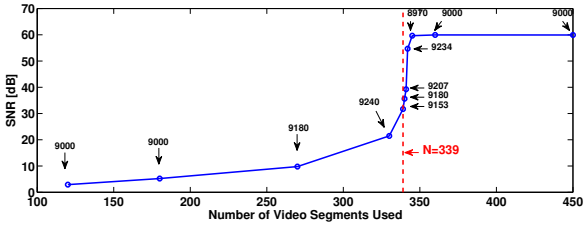
To sum up, for the first example, we have shown two useful bounds for perfectly identifying the projection $\mathcal{P}_1^s \mathbf{h}_1$ of a receptive field $\mathbf{h}_1$ onto a Hilbert space $\mathcal{H}_3^{s|3}$. The first lower bound is that the total number of measurements must be greater or equal to the dimension of the space $dim(\mathcal{H}_3^{s|3})$. Equivalently, the totality of spikes produced in response to M experimental trials involving M different video clips must be greater than $dim(\mathcal{H}_3^{s|3}) + M$. The second lower bound is that the number of video clips M must be greater or equal to $3 \cdot dim_{xy}(\mathcal{H}_3^s)$. Both conditions must be satisfied at the same time. In addition, we see that each video clip can provide a maximum of $2L_t + 1$ informative measurements towards the



(a)



(b)



(c)

Figure 5.3: Effect of the number of video clips and the total number of measurements on the quality of identification. (a) A fixed number of measurements (19, 25 and 29) is used from each video clip. SNR increases as more video clips are used until the total number of measurements passes the dimension of the space (19, 25) and the number of video clips reaches $3 \cdot dim_{xy}(\mathcal{H}_3^s)$ (25,29) (b) A fixed number of video clips are used in identification while the number of measurements from each video clip increases. The identification performance saturates after each video clip generates 25 measurements. This number corresponds to $dim_t(\mathcal{H}_3^s)$. This suggests that a minimum of 26 spikes is needed to fully explore the space in the time dimension.

Figure 5.3: (Continued) (c) Varying the number of video clips in identification while keeping the number of total measurements above $dim(\mathcal{H}_3^{s|3})$ at all times, the identification performance steeply increases after the number of video clips reaches $3 \cdot dim_{xy}(\mathcal{H}_3^s)$. This suggests that in order to identify the color receptive field, not only the total number of spikes has to be larger than $dim(\mathcal{H}_3^{s|3})$, but also the number of video clips used needs to be larger than $3 \cdot dim_{xy}(\mathcal{H}_3^s)$. (Modified from [118] with permission, ©2015 Elsevier)

identification.

We now consider using a long sequence of continuous natural video in identifying the entire neural circuit. Colors in natural visual scenes are much more correlated than in randomly generated bandlimited signals using the above procedure. As neural systems are tuned to the natural statistics, it is likely that neurons will respond differently to natural stimuli. Thus, there is a need to be able to accommodate the use of natural stimuli during functional identification in real experiments. The machinery of RKHS, and spaces of trigonometric polynomials specifically, provide that capability. It is essential, however, to properly segment a long natural sequence into multiple segments with appropriate spike times associated with each. This is detailed in appendix C.3.

In examples below we use a custom natural video shot by a handheld device. The total length of the video is 200 seconds. We use this single video to identify the complete neural circuit, that is, the receptive fields of all $M = 30,000$ neurons. Due to computational constraints and in the interest of time, we identify each of the receptive field components $h_{1|m}^i(x, y, t)$ separately rather than the entire $\mathbf{h}_1^i$. This can easily be done by supplying a single color channel during the identification procedure.

For simplicity, we assume that the dilation parameter of the receptive field was known. For $\alpha = 2^{0.5}$, the chosen screen size is $24\,[\text{px}] \times 24\,[\text{px}]$, $\Omega_x = \Omega_y = 0.5$. For $\alpha = 2^{1.5}$, the chosen screen size is $48\,[\text{px}] \times 48\,[\text{px}]$ and $\Omega_x = \Omega_y = 0.25$. In both cases, $L_x = L_y = 12$, $L_t = 4$ and $\Omega_t = 2\pi \cdot 20$. The dimension of both spaces is $\dim_{xy}(\mathcal{H}_3^s) \times (2L_t + 1) = 3,969$, where $\dim_{xy}(\mathcal{H}_3^s) = 441$. In this example, each neuron in the population has fixed but different parameters and generates about 100 spikes per second, or about $10 = 2L_t + 2$ spikes per windowed video clip. Note here that since we only use spikes generated in the

second half of the $S_t = 0.2$ window (see Figure C.1(a)), the number of spikes used per windowed video clip is approximately $100 \times S_t/2 = 10$. This choice of stimulus and neuron parameters allows each neuron to provide the maximum number of informative spikes about each video clip in the simulation. We vary the number of spikes used in the identification. The number of video clips co-vary with the number of spikes as a result. Each receptive field is identified on a single GPU and it takes about 10 minutes when using 17,000 spikes to identify the receptive field. When using only 1,000 spikes, it takes 30 seconds to finish the identification process. Since a large number of neurons had to be identified, we performed simulation on a cluster that has 96 GPUs.

The SNR of the identified receptive fields over the original receptive fields are shown in Figure 5.4, where different colors are used to indicate a different number of total measurements used in identification. Each dot in the figure corresponds to the SNR of an identified receptive field for the corresponding neuron. Figure 5.4 shows a general trend that a larger number of measurements produces better identification results.

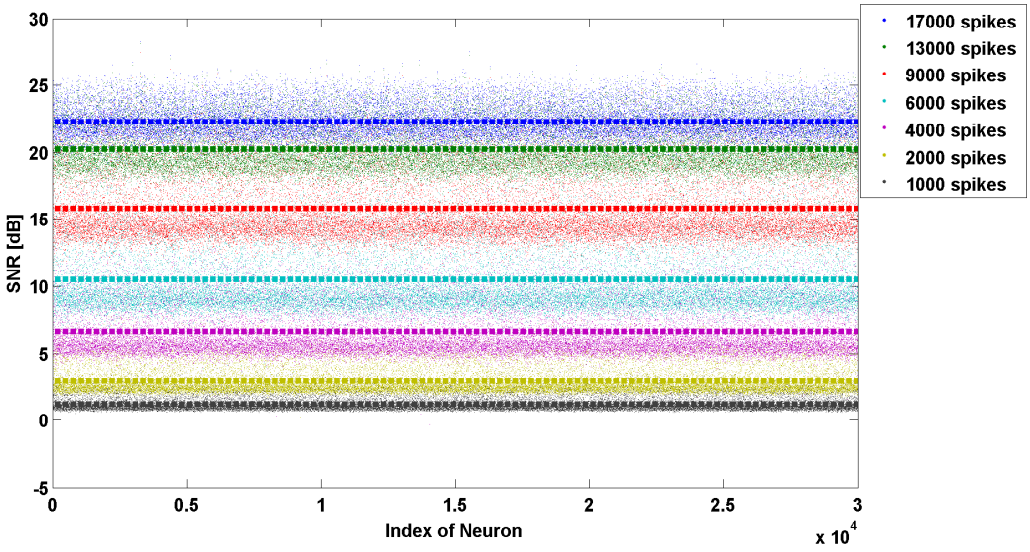


Figure 5.4: SNR of the 30,000 identified filters. Color indicates the number of total measurements used in the identification for each receptive field. The average SNR is shown by dashed line for corresponding colors. (Modified from [118] with permission, ©2015 Elsevier)

5.2 Jointly Evaluating Encoding, Decoding and Identification

Functional identification of a visual neural circuit provides a quantitative description of the relationship between the input video and the spiking activity of the neurons in the circuit. It is natural, then, to ask if this description can be used to reconstruct the actual stimulus that produces the spikes.

By connecting the CIM and TEM methodologies and exploiting their duality we provide an answer to this question in Section 5.2.1. After functionally identifying the entire neural circuit, we apply a novel visual stimulus to the identified encoding circuit. From the resulting spike trains the visual stimulus is recovered using the identified circuit parameters (of the receptive fields). As shown in Section 5.2.2 this provides a single measure for the quantitative evaluation of the identification process.

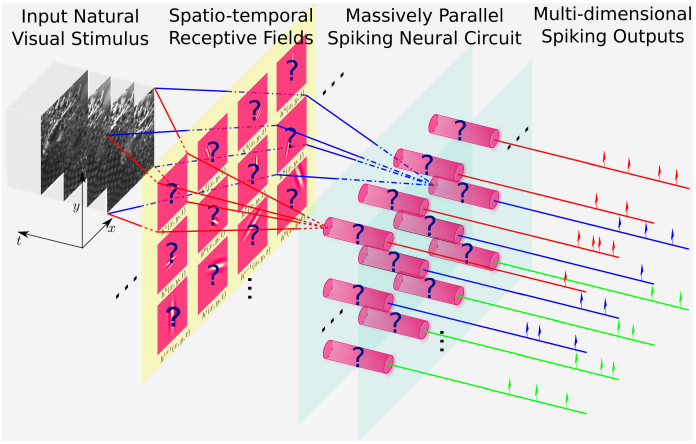
5.2.1 Stimulus Reconstruction Using the Identified Circuit

Decoding of a visual stimulus from spikes produced by a population of identified visual neurons was previously investigated under various contexts. [192] attempted to decode the time course of stimuli with spatially uniform intensity by identifying an optimal linear decoder for retinal ganglion cells in salamanders. A similar decoding algorithm was used to decode natural visual scenes from neural activity in the cat LGN [177]. Both of these approaches are based on linear decoding using the firing rate of the neurons. In addition, only a small number of neurons was considered.

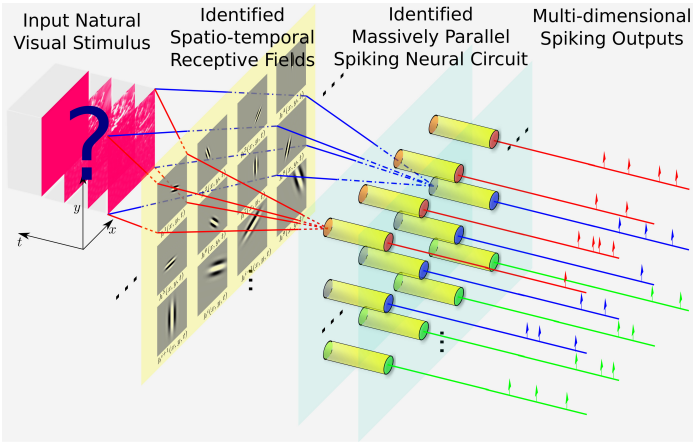
Our approach differs in three important ways. First, instead of decoding from firing rate of neurons with a linear decoder, we use a non-linear decoder applied to the exact spike times generated by the neurons. This is attractive from an experimental standpoint, since the stimulus can be recovered using a single experimental trial, eliminating the need to repeat the same experiment in order to compute the firing rate. The latter may not be possible due to the variability in experimental conditions and the state of the neural circuit. Second, the neural circuit we are investigating is a massively parallel neural circuit comprised of thousands of neurons. In recent years it has become apparent that information in the brain is in general represented by vast populations of neurons that process sensory

stimuli in a parallel fashion. The required population size is both modality- and stimulus-specific and our findings provide an estimate for the lower bound on the number of neurons that are needed to faithfully represent a color video stimulus in the spike domain. Finally, we have provided conditions under which the neural circuit can be identified and the input videos can be subsequently faithfully reconstructed. Note that, in identification, we can only identify the projection of the neural circuit parameters onto the RKHS. Therefore, using the identified neural circuit, it is only possible to reconstruct a stimulus up to its projection onto the same RKHS.

Furthermore, by bridging the identification and reconstruction problems, it is possible to evaluate the amount of information encoded by neurons. One approach is summarized in Fig. 5.5. Given a massively parallel neural circuit, we first identify its parameters (Fig. 5.5(a)). This can be done by presenting an input video and using the resulting spike train in identification. Second, for a novel stimulus presented to the same neural circuit, the spike train generated by the neural circuit can be used along with the identified parameters to reconstruct the novel stimulus (Fig. 5.5(b)).



(a)



(b)

Figure 5.5: Evaluation of massively parallel neural circuit. (a) Neural circuit identification. Known stimuli are presented to a neural circuit whose parameters are unknown, the circuit needs to be identified from the stimuli and the spike times generated. (b) Diagram of the encoding circuit. An unknown stimulus is presented to the circuit. The circuit has been identified and spikes are known, the visual stimulus can be reconstructed from the identified parameters and the spike times. (Modified with permission from: [110], ©2014 IEEE)

5.2.2 Evaluation of Functional Identification in the Stimulus Space

There are three ways to evaluate the quality of the identification process, as schematically illustrated in Fig. 5.6. In simulation, the original receptive fields are known, and we can compare the identified circuit to the original (indicated by (1) in Figure 5.6). Such evaluation is performed in the *parameter space*. The drawback of this approach is that in biological neural systems the ground truth is not known. In other words, there is no true parameter space to compare with.

An alternative and widely adopted approach is to use the identified circuit to predict the response of the circuit to novel stimuli (indicated by (2) in Figure 5.6). A novel stimulus is presented to both the neural circuit and the identified circuit and their responses are compared. In other words, this type of verification is performed in the *spike train space*. In this approach the identified circuit can be viewed as an I/O equivalent or phenomenological circuit. Often, however, such I/O equivalence cannot be made on a spike by spike basis. Typically, only coarser measurements such as the Peri-Stimulus Time Histogram (PSTH) are available for making predictions [22]. Although these coarser metrics are often good useful indicators of the identification performance, they exhibit several shortcomings, especially in population encoding and natural stimuli settings. When evaluating the identification performance of a massively parallel neural circuit, checking the PSTH for every neuron alone typically amounts to thousands of measurements. Moreover, quantitative distance measures between PSTHs are often times not well defined and are hard to interpret. Most importantly, individual PSTH predictions may not speak for the overall functionality of the entire identified circuit.

Within the framework of time encoding and channel identification, we introduce a third way of directly assessing the quality of the identification using the information content retained in the spike train. This metric is the distance between the original visual stimulus and the stimulus decoded using the identified circuit parameters (indicated by (3) in Fig. 5.6). When using the identified filters in decoding instead of the underlying filters, the t-transform corresponding to (4.13) is no longer precise since the identified filters deviate from the projections of the underlying filters. However, it is expected that the error in the t-transform

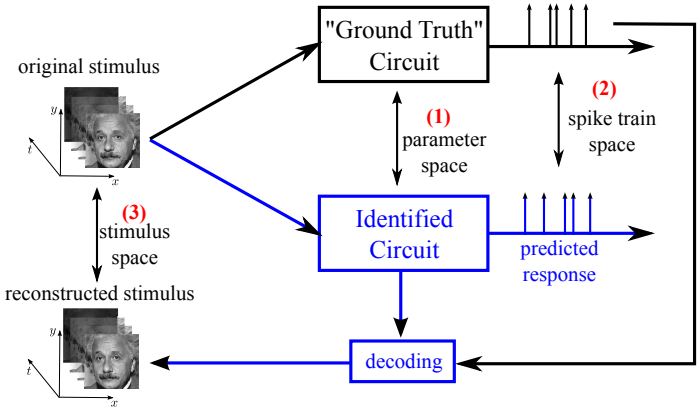


Figure 5.6: Three ways of evaluating the identification quality. (Modified from [118] with permission, ©2015 Elsevier)

should decrease when the identification quality increases, thereby leading to improved reconstruction. Similarly, a poor quality of the decoded cross-validating stimulus suggests that the identified circuit parameters deviate from their true values. By using large scale simulations, the converse can be observed, *i.e.*, a high quality reconstruction indicates a high quality functional identification.

By evaluating the identified circuit in the *stimulus space*, we obtain a metric that quantifies the identification result. In other words, the evaluation of the entire identified neural circuit is reduced to intuitive comparisons in the stimulus space, thereby augmenting the usual neuron-by-neuron comparison in the spike-train space. For images and video, such an evaluation can be visually performed. In addition, many well defined distance measures are readily available in the signal processing literature (see [190] and references therein).

This methodology of using identified parameters in decoding may allow us to identify a relevant subset of neurons in a large scale neural circuit and, at the same time, evaluate the goodness of the identification algorithm for that neuronal subset. In the ideal case, the entire circuit can be identified. We use here faithful representation as an example. By bridging the identification problem (channel identification) with the encoding problem (time encoding), we are able to evaluate the function of the entire circuit by decoding the stimulus and quantifying the information content retained in the spikes.

5.2.3 Joint Performance Evaluation

To illustrate the results of the methodology presented in the previous section, we performed computer simulations of encoding, decoding and identification using the massively parallel neural circuit described in Section 4.3.2 and identified in Section 5.1.2. We also evaluated the performance of the identification algorithm in the stimulus space. Here we present the results obtained and discuss practical issues arising in using the TEMs, TDMs and CIMs with natural videos.

In Section 5.1.2, we identified a massively parallel neural circuit with $30,000$ neurons. The identified receptive fields were compared to the ground truth in the parameter space (see also Figure 5.6), and it was shown that the quality of the identified receptive fields improves with the length of the test stimulus. Using the identified receptive fields, we evaluate the quality of functional identification in the stimulus space.

The neural circuit is first identified with 7 different settings. In each setting, a stimulus of a given duration was used for identifying the receptive fields (see also Figure 5.4). As the length of the stimuli increases, more spikes are obtained for identification Using each set of identified parameters, we decode the spikes generated by the Color Video TEM when encoding novel stimuli. Since a perfect reconstruction of the video can be demonstrated for the case when the underlying receptive fields are known (see Section 4.3.2), we know that it is possible to reconstruct the stimulus with high quality if the circuit is identified well. The decoding procedure, including the parameter of the space and the stitching window, was taken to be exactly the same as that in Section 4.3.2. The decoding using identified neural circuit parameters takes the same time as the decoding in Section 4.3.2, that is, about 2 days for 10 seconds of video.

The reconstruction quality is shown in the video in Figure 5.7. The SNR of the decoded video increases as more measurements are used for identifying each receptive field in the neural circuit. This indicates a better overall quality in the identification step. Such an evaluation in the stimulus space is also consistent with the parameter space evaluation in Section 5.1.2 that is based on the ground truth neural circuit. Moreover, the reconstructed videos are visually self-explanatory. The reconstruction artifacts are clearly visible in the

videos to the left of the vertical line that marks the theoretical lower bound on the number of measurements required for identification. Those videos to the right of the vertical line have a superior quality and are visually close to what can be reconstructed when using the underlying receptive fields.

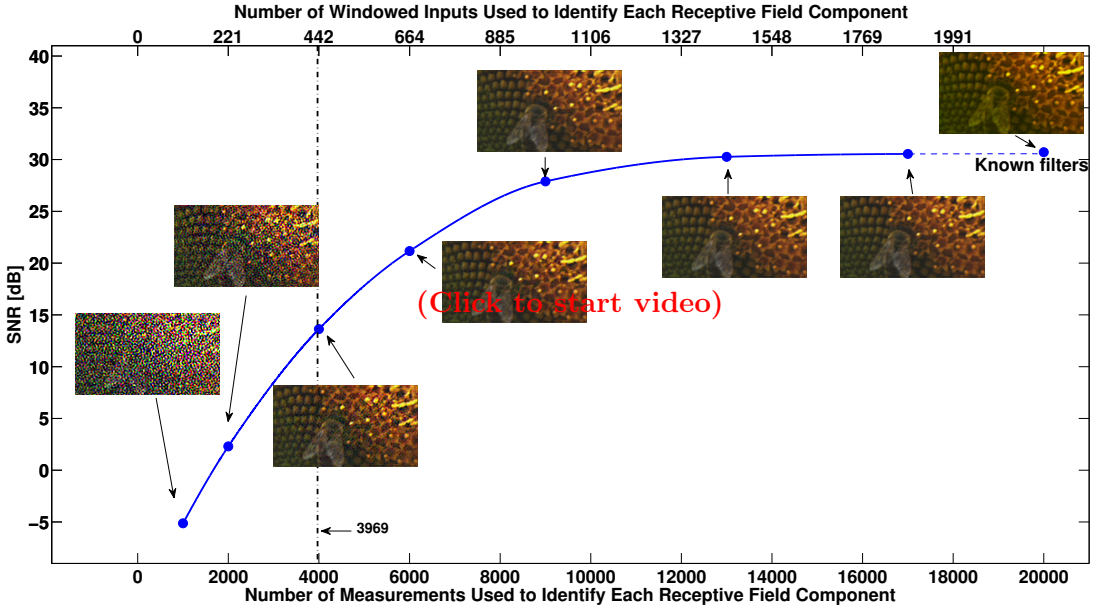


Figure 5.7: Evaluation of identification in the stimulus space. Neural circuit identified with different number of measurements is used to reconstruct a novel stimulus. The increase of SNR of the reconstructed video shows the improved identification quality as the number of measurements used in identifying each receptive field in the neural circuit increases. Snapshots of the corresponding reconstruction are shown with their SNR. (Modified from [118] with permission, ©2015 Elsevier)

Comparing to the evaluation in Section 5.1.2, the quality assessment in the stimulus space appears to be much more straightforward and more intuitive. It also provides a measure of the encoding capability of the identified circuit. Specifically, one can readily evaluate what information about the stimulus is preserved by the neural circuit by simply comparing the original video and the video decoded using the identified circuit parameters.

5.3 Summary

We have presented a comprehensive, yet highly intuitive method for evaluating the functional identification of massively parallel neural circuits. The problem was formulated in the space of color visual scenes, but can generally be applied to other stimulus spaces, *e.g.*, monochrome videos, or other sensory modalities.

The key result that led to the evaluation of identification algorithms in the stimulus space is the duality between

- the decoding problem of a single stimulus encoded by a bank of filters in cascade with a population of neurons, and
- the identification problem of a single receptive field using different stimuli in multiple experimental trials.

The evaluation of functional identification using the proposed method showed that the algorithm for identifying component DSPs was highly effective when applying natural stimuli. It is also important to note that only the projection of the DSPs onto the stimulus space can be properly identified.

We have implicitly assumed that the parameters of the spike generator are always known, even in the functional identification setting. This may not hold true in practice and additional experiments may be required to estimate these parameters before the identification of the receptive field [103]. However, evaluation in the stimulus space is still applicable to assess the identification quality of both filters and neurons.

Chapter 6

Volterra Dendritic Stimulus Processors and Biophysical Spike Generators with Intrinsic Noises Sources

Intrinsic noise sources are diverse and appear on many levels of a neuronal system ranging from electrical to chemical noise sources [40, 52] and from single cells to networks of neurons. At the cellular and subcellular level, variability in biochemical reactions leads to stochastic transduction processes [176], and ion channel fluctuations [138, 196] result in variability in spike generation and propagation [50]. At the network level, probabilistic quantal release of neurotransmitters [85], background synaptic activity [39, 82] and variability in timing of spikes from presynaptic neurons [51] are sources of stochastic fluctuation of synaptic conductances [38] that are believed to have a major impact on spike time variability [201].

The existence of sources of noise also leads to variability in the spike times even when neurons are subject to the same, repeated inputs [11, 18, 37]. Spikes are the primary form of carriers of information in the nervous system and their timing is thought to be relevant

to the message neurons need to convey [161]. Therefore, the variability of spike timing may reduce or damage the information being transmitted. It is quite remarkable, however, that sensory systems manage to be very robust even if they are subject to interference due to noise. Visual and auditory systems are two examples in which the stimuli are highly time varying. These systems have been reported to convey information with high spike timing precision [17, 86].

Noise may be useful in facilitating signal detection [133]. Still, interference due to noise poses an important limit on how well sensory systems can represent input stimuli. It is not clear how intrinsic noise sources affect the representation of sensory inputs based on spike times, and how they impact the functional identification of sensory neurons.

We study the representation of sensory stimuli using a novel neural circuit model, that extends previously proposed models [104, 105, 112] in terms of architectural complexity and the existence of intrinsic noise sources. Our base level circuit architecture consists of two interconnected neurons, each with two cascaded stages. The first stage comprises two types of dendritic stimulus processors. The first dendritic stimulus processor performs *nonlinear* processing of input stimuli in the feedforward path leading to the spike generator. The second dendritic stimulus processor performs *nonlinear* processing in the feedback loop whose inputs are spike trains generated by biophysical spike generators (BSGs). The BSGs constitute the second stage of the base level circuit.

Our nonlinear dendritic stimulus processors describe functional I/O relationships between the dendritic outputs in the first stage and inputs that are either sensory stimuli or spikes generated by BSGs. DSPs are modeled using Volterra series. Volterra series have been used for analyzing nonlinear neuronal responses in many contexts [43, 126], and have been applied to the identification of single neurons in many of sensory areas [9, 30, 182]. Volterra dendritic processors can model a wide range of nonlinear effects commonly seen in sensory systems [105]. Here, in addition, we introduce nonlinear interactions between neurons in feedback and cross-feedback paths. This gives rise to interesting neural processing capabilities directly in the spike domain, *e.g.*, coincidence detection [2, 179]. The relationships described here by the Volterra model are functional and do not address the underlying circuit/dendritic tree level interactions. However, the latter have recently been subject to

intense investigations [124, 194, 198, 200, 202, 204]. Conductance-based, biophysical spike generators are well established models that have been extensively used in studies of neuronal excitability and in large simulations of spiking neural networks [81]. Following [95], we use formal BSG models to represent sensory stimuli under noisy conditions.

We formulate the encoding, decoding and functional identification problems under the neural encoding framework of Time Encoding Machines (TEMs). In this modeling framework the exact timing of spikes is considered to carry information about input stimuli [106]. The separation into dendritic stimulus processors and spike mechanisms mentioned above allows us to study synaptic inputs and spike generation mechanisms separately, and hence independently model the intrinsic noise sources of each component. We incorporate two important noise sources into a general single-input multi-output neural circuit model. The first is a channel noise source that arises in spike generation [197]. The second is a synaptic noise source due to a variety of fluctuating synaptic currents [128].

Based on the rigorous formalism of TEMs, we show how noise arising in dendritic stimulus processors and in biophysical spike generators is related to the measurement error in generalized sampling. Dendritic stimulus processing and spike generation can then be viewed as a generalized sampling scheme that neurons utilize to represent sensory inputs [112]. Contrary to traditional sampling where the signal amplitude is sampled at clock times, neurons asynchronously sample all stimuli.

We systematically investigate how the strength of noise sources degrades the faithfulness of stimulus representation and the quality of functional identification of our proposed class of neural circuits. Furthermore, since the representation is based on spike timing, it is natural to investigate how spike timing variability affects the precision in representing the amplitude information of sensory stimuli.

There are a number of benefits in the investment of mathematical formalism employed here. Formulating the problem of stimulus encoding with a neural circuit with intrinsic noise sources as one of generalized sampling, *i.e.*, of taking noisy measurements is of interest to both experimentalists and theoreticians alike. Understanding that the problem of neural decoding and functional identification are dual to each other is key to building on either or both. Finding how many repeat experiments need to be performed for a precise quantitative

identification of Volterra kernels is of great value in neurophysiology. A further qualitative insight of our work is that for neural circuits with arbitrary connectivity, feedforward kernels are typically easier to estimate than feedback kernels. Finally, our finding that some key non-linear neural circuits are tractable for detailed noise analysis suggests a wide reaching analytical methodology.

6.1 Modeling Nonlinear neural circuits, Stimuli and Noise

We present in Section 6.1.1 the general architecture of the neural circuits considered in this chapter. In Section 6.1.2 we discuss the modeling of the space of stimuli. Volterra DSPs are the object of Section 6.1.3. Finally, in Section 6.1.4 we provide models of BSGs with intrinsic noise sources.

6.1.1 Neural circuit architecture

The general architecture of the neural circuit considered here is shown in simplified form in Figure 6.1 [109]. It consists of two neurons with a common time-varying input stimulus. With added notational complexity the neural circuit in Figure 6.1 can easily be extended in two ways. First, multiples of such circuits can encode a stimulus in parallel, as shown in Figure 6.2. In this case only pairs of neurons are interconnected through the feedback kernels. Second, more neurons can be considered in the neural circuit of Figure 6.1; all these neurons can be fully interconnected through feedback loops.

Each neuron $i, i = 1, 2$, receives a single time-varying input stimulus $u_1(t)$. The modeling of the input stimulus is discussed in Section 6.1.2. The output of each of the biophysical spike generators (BSGs) is a spike sequence denoted by (t_k^i), $k \in \mathbb{Z}$, and $i = 1, 2$.

The input stimulus $u_1(t)$ is first processed by a feedforward Dendritic Stimulus Processor (feedforward DSP) [105]. The feedforward DSP models the aggregated effect of processing in the neural circuits in the prior stages and in the dendritic tree of neuron $i = 1, 2$. For example, if the neurons in the model circuit are considered to be Retinal Ganglion Cells (RGCs), then the feedforward Volterra DSP models the processing that takes place in the outer- and inner-plexiform layers of the retina as well as in the dendritic trees of an RGC

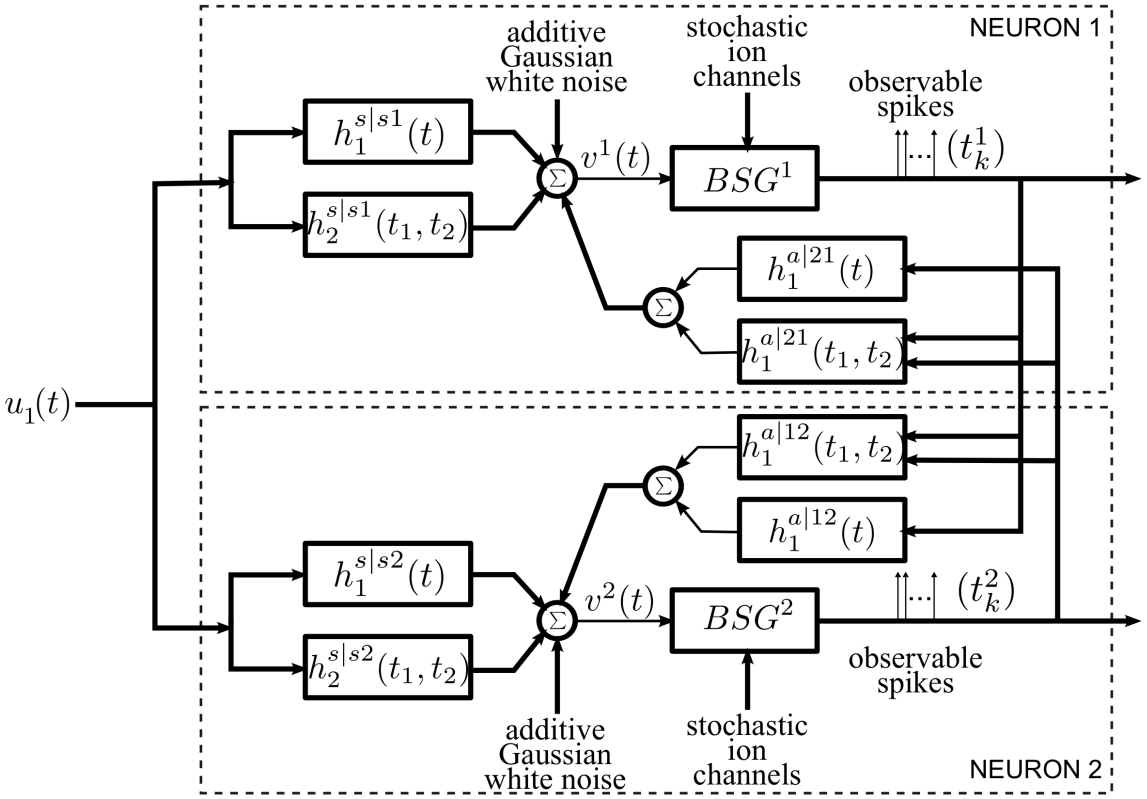


Figure 6.1: Diagram of the architecture of the neural circuits.

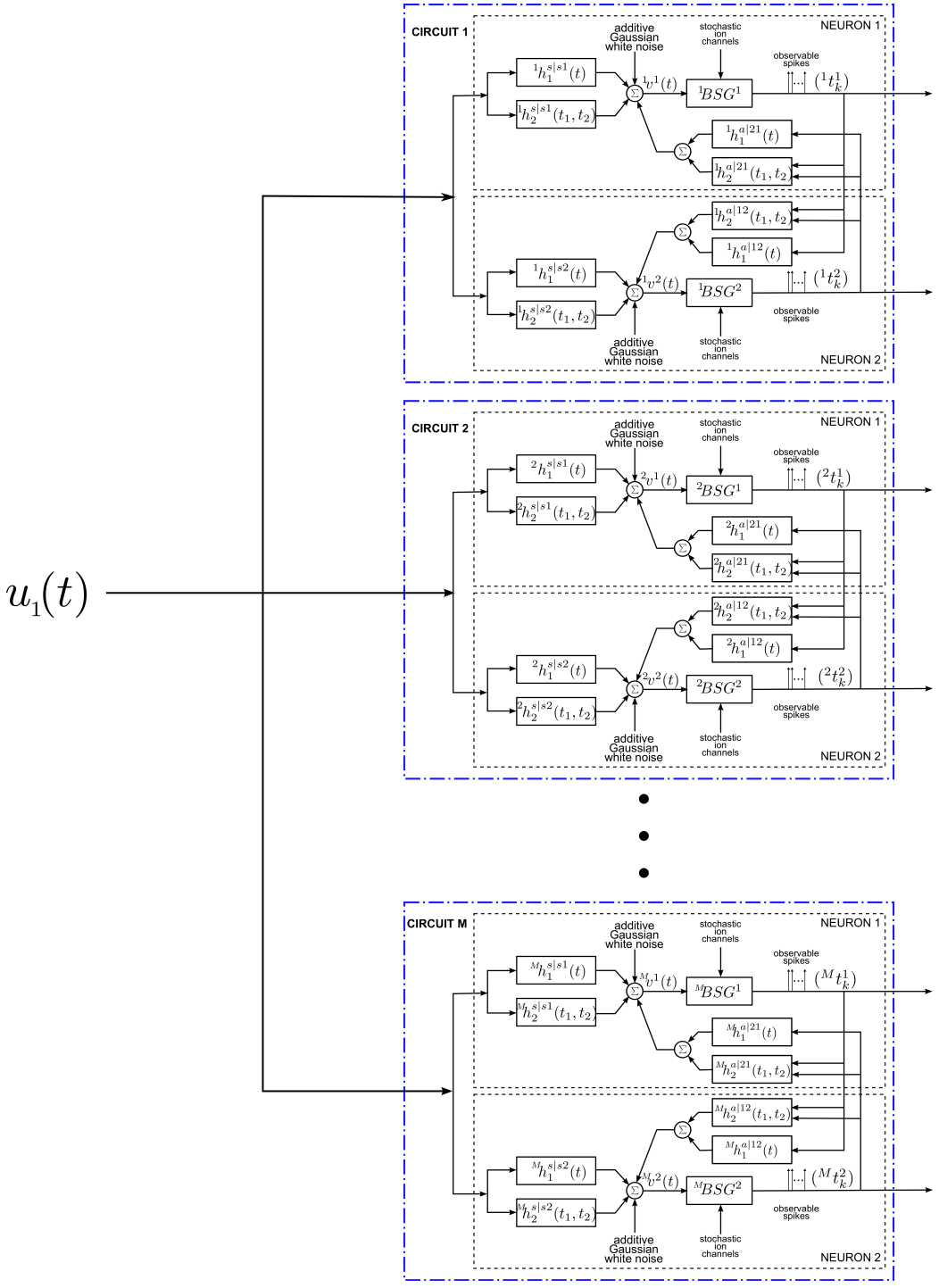


Figure 6.2: Single-Input Multi-Output (SIMO) encoding diagram with, in parallel, M of the neural circuits in Figure 6.1.

[131, 194]. The feedforward DSPs are modeled here as second order Volterra expansion terms [187]. The first order terms $h_1^{s|si}(t)$ in the feedforward DSPs are linear filters typically used in modeling receptive fields. The second order terms $h_2^{s|si}(t_1, t_2)$ model nonlinear operations on the stimulus $u_1(t)$.

A second group of Volterra DSPs models the cross-feedback interactions between the two neurons. Instead of time-varying stimuli, the output spikes generated by the BSGs are the inputs to these DSPs. We therefore refer to these as feedback Dendritic Stimulus Processors (feedback DSPs). The output spikes of each individual neuron i are processed by the first order term $h_1^{a|ji}(t), i, j = 1, 2, i \neq j$. In addition, output spikes from both neurons interact nonlinearly through the second order terms $h_2^{a|ji}(t_1, t_2), i, j = 1, 2, i \neq j$. The summed responses from the first order feedback DSP $h_1^{a|ji}$ and the second order feedback DSP $h_2^{a|ji}$ are fed back to neuron i as additional dendritic currents.

The dendritic currents consisting of the output of the DSPs with added noise are subsequently encoded by biophysical spike generators. BSGs are biophysically realistic axon hillock spike generator models that are governed by a set of differential equations with multiple types of ion channels [76, 81]. The detailed BSG models are introduced in Section 6.1.4. The spike times of output spikes generated by the BSGs are assumed to be observable.

We identify two intrinsic noise sources of the proposed neural circuit. First, the feedforward DSPs and the feedback DSPs are affected by additive Gaussian white noise. This noise arises from the combined effect along the path from sensory transduction to synaptic integration and includes synaptic background noise and stochasticity in the dendritic tree [40, 54, 128]. Since the outputs of the feedforward and feedback DSPs are additively combined, we consider, for simplicity, a single source of additive Gaussian white noise. Second, the ion channels of the BSGs are intrinsically stochastic and introduce noise in the spike generators [74, 197].

6.1.2 Modeling Signal Spaces

Two signal spaces will be considered here. The first, models the space of input signals to feedforward DSPs. The second models the space of input spikes to feedback DSPs. These spaces will be formally described below.

6.1.2.1 Modeling the Space of Input Stimuli

We model the space of input stimuli as a Reproducing Kernel Hilbert Space (RKHS) [10]. RKHSs are versatile vector spaces for modeling signals arising in computational neuroscience, signal processing and machine learning. For example, auditory signals, olfactory signals and visual signals can readily be modeled as band-limited functions of an RKHS with a sinc or Dirichlet kernel [100, 112]. A particular choice of RKHSs in this article is the space of trigonometric polynomials. The computational advantage of working on the space of trignometric polynomials has been discussed [112] and is closely related to the algorithmic tractability of the Fourier series in the digital domain. If the biological signals have unknown bandwidth with a spectrum that falls off fast enough, many Sobolev spaces might be a suitable choice of RKHS [10, 97]. In such spaces the norm may include the derivative of the signal, *i.e.*, the rate of change of the signal that many neurons are sensitive to [89].

The space of trigonometric polynomials is defined as below.

Definition 8. *The space of trigonometric polynomials $\mathcal{H}_1^s$ is a function space whose elements are functions defined on the domain $\mathbb{D}_1 = [0, S_t^s], S_t^s \in \mathbb{R}_+$, of the form*

$$u_1(t) = \sum_{l=-L_t^s}^{L_t^s} u_l e_l(t), \tag{6.1}$$

where

$$e_l(t) = \frac{1}{\sqrt{S_t^s}} e^{jl\frac{\Omega_t^s}{L_t^s}t}, l = -L_t^s, \cdots, L_t^s, \tag{6.2}$$

are a set of orthonormal basis functions. Ω_t^s denotes the bandwidth and L_t^s is the order of the space.

$\mathcal{H}_1^s$ endowed with the inner product:

$$\langle u_1, v_1 \rangle = \int_{\mathbb{D}_1} u_1(t)\overline{v_1(t)}dt \tag{6.3}$$

is a Hilbert Space. Intuitively, the basis functions $e_l(t), l = -L_t^s, \ldots, L_t^s$, can be interpreted as a set of discrete spectral lines uniformly spaced in the frequency domain between $-\Omega_t^s$ and Ω_t^s. For a given signal $u_1(t)$, the amplitude of its spectral lines is determined by the coefficients $u_l, l = -L_t^s, ..., L_t^s$.

$\mathcal{H}_1^s$ is an RKHS with reproducing kernel (RK)

$$K_1^s(s;t) = \sum_{l=-L_t^s}^{L_t^s} e_l(t-s). \tag{6.4}$$

It can be easily verified that the RK satisfies the reproducing property

$$\langle u_1(\cdot), K_1^s(\cdot;t)\rangle = u_1(t), \forall u_1 \in \mathcal{H}_1^s, t \in \mathbb{D}_1. \tag{6.5}$$

Definition 9. *We shall also consider the tensor product space $\mathcal{H}_2^s = \mathcal{H}_1^s \otimes \mathcal{H}_1^s$ on the domain $\mathbb{D}_2 = [0, S_t^s] \times [0, S_t^s]$, whose elements are of the form*

$$u_2(t_1,t_2) = \sum_{l_1=-L_t^s}^{L_t^s} \sum_{l_2=-L_t^s}^{L_t^s} u_{l_1 l_2} e_{l_1 l_2}(t_1,t_2), \tag{6.6}$$

where

$$e_{l_1 l_2}(t_1,t_2) = \frac{1}{S_t^s} e^{jl_1 \frac{\Omega_t^s}{L_t^s} t_1} e^{jl_2 \frac{\Omega_t^s}{L_t^s} t_2}, \tag{6.7}$$

are a set of functions forming an orthonormal basis.

$\mathcal{H}_2^s$ is again an RKHS with RK

$$K_2^s(s_1,s_2;t_1,t_2) = \sum_{l_1=-L_t^s}^{L_t^s} \sum_{l_2=-L_t^s}^{L_t^s} e_{l_1 l_2}(t_1-s_1, t_2-s_2). \tag{6.8}$$

Note that we use the subscript to indicate the dimension of the domain of functions, *i.e.*, the number of variables the functions in the RKHS have, and use the superscript s to indicate the space associated with input stimuli. Note also that in Section 4, we denote the color components by u_1, u_2 and u_3. Here, by abuse of notation, u_1 and u_2 defined here are not associated with color video. Rather, u_1 denotes the stimulus that have been previously denoted as u, and u_2 denotes elements in the tensor product space.

The reason to introduce the tensor product space is that $u_1(t_1)u_1(t_2) \triangleq u_2(t_1,t_2)$ is an element in $\mathcal{H}_2^s$. We shall see later in this chapter that the processing of the second order Volterra DSP, while nonlinear on u_1, is linear on u_2 [105].

Projections of functions onto the RKHSs introduced here can be defined as follows:

Definition 10. *Let $h_1 \in \mathbb{L}^1(\mathbb{D}_1)$, where $\mathbb{L}^1$ denotes the space of Lebesgue integrable functions. The operator $\mathcal{P}_1^s : \mathbb{L}^1(\mathbb{D}_1) \to \mathcal{H}_1^s$ given by*

$$(\mathcal{P}_1^s h_1)(t) = \int_{\mathbb{D}_1} h_1(s) K_1^s(s;t) ds,$$

is called the projection operator from $\mathbb{L}^1(\mathbb{D}_1)$ to $\mathcal{H}_1^s$.
Similarly, let $h_2(t_1, t_2) \in \mathbb{L}^1(\mathbb{D}_2)$, the operator $\mathcal{P}_2^s : \mathbb{L}^1(\mathbb{D}_2) \to \mathcal{H}_2^s$ (by abuse of notation) given by

$$(\mathcal{P}_2^s h_2)(t_1, t_2) = \int_{\mathbb{D}_2} h_2(s_1, s_2) K_2^s(s_1, s_2; t_1, t_2) ds_1 ds_2,$$

is called the projection operator from $\mathbb{L}^1(\mathbb{D}_2)$ to $\mathcal{H}_2^s$.

6.1.2.2 Modeling the Space of Spikes

The feedback kernels of the neural circuit in Figure 6.1 receive as inputs spike trains generated by the BSGs. Spike trains are often modeled as sequences of Dirac delta pulses and, consequently, the outputs of linear feedback kernels are the result of superposition of their impulse responses [87, 112, 149].

Dirac delta pulses have infinite bandwidth. Spikes generated by the BSGs, however, have limited effective bandwidth. Following [103] spikes are modeled to be the RK of an one-dimensional Hilbert space $\mathcal{H}_1^a$ at spike time occurrence. Here $\mathcal{H}_1^a$ is a space of trigonometric polynomials whose order L_t^a, period S_t^a and bandwidth Ω_t^a may differ from the input stimulus space $\mathcal{H}_1^s$, where Ω_t^a shall be larger than the bandwidth assumed for the feedback kernel, and S_t^a is much larger than the support of the feedback kernel [103]. A spike at time t_k^i of neuron i can then be expressed in functional form as $K_1^a(t_k^i; t)$, where the superscript indicates that the RK belongs to the spike input space.

Due to the reproducing property, single or pairs of input spikes have the property

$$h_1(t) * K_1^a(t_k^i; t) = \int_{\mathbb{D}_1} h_1(s) K_1^a(t_k^i; t-s) ds = (\mathcal{P}_1^a h_1)(t - t_k^i)$$

and

$$\int_{\mathbb{D}_2} h_2(s_1, s_2) K_2^a(t_k^i, t_l^j; t-s_1, t-s_2) ds_1 ds_2 = (\mathcal{P}_2^a h_2)(t - t_k^i, t - t_l^j)$$

for $i, j = 1, 2, i \neq j$. The operator $\mathcal{P}_1^a$ and $\mathcal{P}_2^a$ are similarly defined to $\mathcal{P}_1^s$ and $\mathcal{P}_2^s$, respectively, in Definition 10; they denote, however, the projections onto the space of spikes. Thus, not

surprisingly, incoming spikes directly readout the projection of the feedback kernels. By letting $L_t^a \to \infty$, $(\mathcal{P}_1^a h_1)(t-t_k)$ shall converge to $h_1(t-t_k)$ in $\mathbb{L}^2$ norm as the RK converges to the *sinc* function and the RKHS becomes the space of band-limited signals [112]. A more detailed analysis is available in [103]. This formalism will be employed for solving the functional identification problem formulated in Section 6.3.1.

6.1.3 Volterra Dendritic Stimulus Processors

As mentioned in Section 6.1.1, two forms of dendritic stimulus processing appear in our model.

6.1.3.1 Feedforward Volterra Dendritic Stimulus Processors

The feedforward DSPs are modeled as up to second order terms in the Volterra series. The feedforward DSPs take continuous signals in the stimulus space as inputs, while the output can be expressed as (see also Figure 6.1)

$$\int_{\mathbb{D}_1} h_1^{s|si}(t-s)u_1(s)ds + \int_{\mathbb{D}_2} h_2^{s|si}(t-s_1,t-s_2)u_1(s_1)u_1(s_2)ds_1ds_2, \tag{6.9}$$

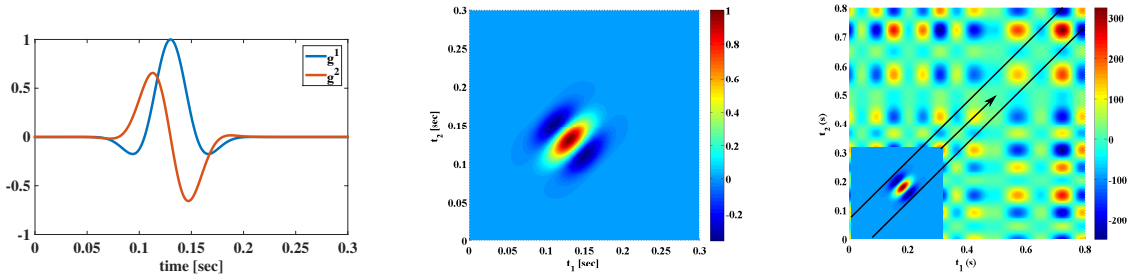
where $h_1^{s|si} \in \mathbb{L}^1(\mathbb{D}_1)$ and $h_2^{s|si} \in \mathbb{L}^1(\mathbb{D}_2)$ denote, respectively, the first and second order Volterra kernels, $i = 1, 2$. They are assumed to be real, causal and bounded-input bounded-output (BIBO)-stable. It is also assumed that both $h_1^{s|si}$ and $h_2^{s|si}$ have finite memory. In addition, $h_2^{s|si}$ is assumed, without loss of generality, to be symmetric, *i.e.*, $h_2^{s|si}(t_1,t_2) = h_2^{s|si}(t_2,t_1)$.

Example 2. *We present here a Volterra DSP that is akin to a model of dendritic stimulus processing of complex cells in the primary visual cortex (V1). The difference is that the complex cells operate spatio-temporally, whereas in the example given below they operate temporally. We first consider two first order kernels based on Gabor functions,*

$$g^1(t) = \exp\left(-\frac{(t-0.13)^2}{2\cdot 0.0005}\right)\cos(2\pi\cdot 10\cdot(t-0.13)),$$

$$g^2(t) = \exp\left(-\frac{(t-0.13)^2}{2\cdot 0.0005}\right)\sin(2\pi\cdot 10\cdot(t-0.13)).$$

The two filters are Gaussian modulated sinusoids, that are typically used to model receptive fields of simple cells in the primary visual cortex (V1) where the variables denote space instead of time [36, 120]. In addition, the two filters are quadrature pair in phase. Both filters are illustrated in Figure 6.3(a). The response of applying the input stimulus u_1 *on the temporal filters with impulse response* g^1 *and* g^2 *is given by* $\int_{\mathbb{D}_1} g^1(t-s)u_1(s)ds$ *and* $\int_{\mathbb{D}_1} g^2(t-s)u_1(s)ds$*, respectively.*



(a) (b) (c)

Figure 6.3: Examples of Volterra kernels. (a) First order kernels of quadrature pair of Gabor functions modeling the receptive fields of simple cells. (b) Second order kernel modeling receptive fields of complex cells. (c) The mechanics of the two dimensional convolution operation between the u_2 ($S_t^s = 0.8, \mathbb{D}_2 = [0, 0.8]\times[0, 0.8]$) and $h_2^{s|si}$. $u_2(t_1, t_2) = u_1(t_1)u_1(t_2)$ is shown in the background. The inset shows the second order Volterra kernel $h_2^{s|si}$ rotated 180° around origin (see also (b)). $h_2^{s|si}$ is only shown in a restricted domain and is zero elsewhere). For $t = 0.3$, the output of the convolution is the integral of the product of the rotated Volterra kernel and the signal underneath. Since the convolution is evaluated on the diagonal $t = t_1 = t_2$, the second order kernel shifts, as t increases, along the arrow on the diagonal.

The responses of the two linear filters of the complex cell model are squared and summed

to produce the phase invariant measure v^i [22], where

$$\begin{aligned} v^i(t) &= \left[\int_{\mathbb{D}_1} g^1(t-s)u_1(s)ds\right]^2 + \left[\int_{\mathbb{D}_1} g^2(t-s)u_1(s)ds\right]^2 \\ &= \int_{\mathbb{D}_2} g^1(t-s_1)g^1(t-s_2)u_1(s_1)u_1(s_2)ds_1ds_2 + \int_{\mathbb{D}_2} g^2(t-s_1)g^2(t-s_2)u_1(s_1)u_1(s_2)ds_1ds_2 \\ &= \int_{\mathbb{D}_2} \left[g^1(t-s_1)g^1(t-s_2) + g^2(t-s_1)g^2(t-s_2)\right] u_1(s_1)u_1(s_2)ds_1ds_2 \\ &= \int_{\mathbb{D}_2} h_2^{s|si}(t-s_1, t-s_2)u_1(s_1)u_1(s_2)ds_1ds_2, \end{aligned} \tag{6.10}$$

where $h_2^{s|si}(t_1,t_2) = g^1(t_1)g^1(t_2)+g^2(t_1)g^2(t_2)$. Therefore, the operation performed by a complex cell can be modeled with a second order Volterra kernel. $h_2^{s|si}$ is shown in Figure 6.3(b). We now take a closer look at the operation of the second order kernel. The two dimensional convolution of the second order kernel with $u_2(t_1,t_2)$ is shown in Figure 6.3(c).

It is important to note that, since the second order kernel has finite memory, it may not have enough support to cover the entire domain $\mathbb{D}_2$ for $u_2(t_1,t_2)$. For example, as illustrated in Figure 6.3(c), the output of the second order feedforward DSP at time t is given by the integral of the product of $u_2(t_1,t_2)$ and a rotated $h_2^{s|si}$ with the origin shifted to (t,t) (see also (6.10)). Since the shift is along the diagonal, only $u_2(t_1,t_2)$ in the domain that is contained within the black lines is multipled by nonzero values of $h_2^{s|si}$. $u_2(t_1,t_2)$ elsewhere in the domain is always multiplied by zero in evaluating the output. Therefore, the output of the second order filter only contains information about u_2 within the domain located in between the black lines in Figure 6.3(c). This has implications on decoding the signal (see also Remark 11 in Section 6.2.2)

6.1.3.2 Feedback Volterra Dendritic Stimulus Processors

As already mentioned, the feedback DSPs do not operate on stimuli directly but rather on spikes generated by BSGs. We assume that $h_1^{a|ji} \in \mathbb{L}^1(\mathbb{D}_1)$, $h_2^{a|ji} \in \mathbb{L}^1(\mathbb{D}_2)$, $i \neq j$, are real, causal, BIBO-stable and have finite memory. In addition, we assume that these kernels are effectively band-limited (see also Section 6.1.2.2). In functional form we denote a train of

spikes as $\sum_k K_1^a(t_k^i;t)$. The output of the feedback DSP i amounts to

$$\sum_{l\in\mathbb{Z}}(\mathcal{P}_1^a h_1^{a|ji})(t-t_l^j)+\sum_{k\in\mathbb{Z}}\sum_{l\in\mathbb{Z}}(\mathcal{P}_2^a h_2^{a|ji})(t-t_l^j,t-t_k^i) \tag{6.11}$$

with $j\neq i$.

In particular, the inputs to the second order term of the feedback DSPs are generated by two neurons. This allows for modeling nonlinear interactions between the two neurons in the spike domain.

6.1.3.3 Overall Output from DSPs

The overall inputs (without noise) to the two BSGs in Figure 6.1 are

$$\begin{aligned} v^i(t) = &\int_{\mathbb{D}} h_1^{s|si}(t-s)u_1(s)ds+\int_{\mathbb{D}^2} h_2^{s|si}(t-s_1,t-s_2)u_1(s_1)u_1(s_2)ds_1ds_2 \\ &+\sum_{l\in\mathbb{Z}}(\mathcal{P}_1^a h_1^{a|ji})(t-t_l^j)+\sum_{l\in\mathbb{Z}}\sum_{k\in\mathbb{Z}}(\mathcal{P}_2^a h_2^{a|ji})(t-t_l^j,t-t_k^i),\quad \text{and}\quad i,j=1,2,i\neq j. \end{aligned} \tag{6.12}$$

The system of equations (6.12) above functionally describe the post-synaptic aggregate currents that are injected into the BSG i.

There are a variety of noise sources to be considered. Synaptic variability of feedforward DSPs adds noise sources to the current input to the BSGs. These include thermal noise, synaptic background noise, etc. [18, 40, 54, 84, 128]. Feedback DSP kernels may themselves be subject to intrinsic noise sources that may lead to variability in the spike generation process. Intrinsic variability of BSG spike times can, e.g., contribute to the variability of the aggregate current driving the axon hillock in feedback loops.

Overall, the combined effect of DSP noise sources is modeled as Gaussian white noise processes that are added to the feedforward and feedback DSP outputs. The sum total of signal and noise represents the aggregate current input to the BSGs (see Figure 6.1). Formal DSP noise models will be incorporated directly into the BSG model presented in the next section.

6.1.4 Biophysical Spike Generators

6.1.4.1 BSGs and Phase Response Curves

We consider biophysically realistic spike generators as described in Section 3.1.4. The class of BSGs can be expressed in vector notation as

$$\frac{d\mathbf{x}^i}{dt} = \mathbf{f}^i(\mathbf{x}^i, I^i), i = 1, 2, \tag{6.13}$$

where $\mathbf{x}^i$ is a vector of dimension N^i, denoting the state (gating) variables of the neuron model, $\mathbf{f}^i$ is a vector function of the same dimension, and I^i is a constant bias current in the voltage equation of each BSG. As an example, for a Hodgkin-Huxley neuron, $N^i = 4$ (see Appendix A for a detailed description of the Hodgkin-Huxley model).

Each input current $v^i(t)$ is applied to the neuron i by additive coupling to the voltage equation, typically the first of the set of ordinary differential equations, i.e.,

$$\frac{d\mathbf{x}^i}{dt} = \mathbf{f}^i(\mathbf{x}^i, I^i) + \left[v^i(t), \mathbf{0}\right]^T, i = 1, 2, \tag{6.14}$$

where $\mathbf{0}$ is a row vector of appropriate size.

We assume that the neuron is periodically spiking when no external input is applied. This can be satisfied by a constant bias current I^i additively coupled onto the voltage equation. The use of I^i is necessary to formulate the encoding for the single neuron case, and this assumption will be relaxed later in this article.

A large enough bias current induces a periodic oscillation of the biophysical spike generator. Therefore, the phase response curve (PRC) is well defined for this limit cycle [81]. We denote the PRC of the limit cycle induced by the bias current I^i as $\boldsymbol{\psi}^i(t, I^i) = [\psi_1^i(t, I^i), \psi_2^i(t, I^i), \cdots, \psi_{N^i}^i(t, I^i)]^\mathsf{T}$ with appropriate dimension N^i, where $\psi_n^i(t, I^i), n = 1, 2, \cdots, N^i$, are the PRCs associated with the nth state variable. Without loss of generality, we assume that $\psi_1^i(t, I^i)$ is always the PRC associated with the voltage variable.

6.1.4.2 Channel Noise in BSGs

As shown in Figure 6.1, we consider BSGs with noise sources in the ion channels. The noise arises due to thermal fluctuations [74, 197] as the finite number of ion channels in the BSGs open and close stochastically.

The differential equations that govern the dynamics of the BSGs in (6.14) are deterministic. The set of stochastic differential equations (SDEs) below represent their stochastic counterpart [95]:

$$d\mathbf{Y}^i = \mathbf{f}^i(\mathbf{Y}^i, I^i)dt + \mathbf{B}^i(\mathbf{Y}^i)d\mathbf{Z}^i(t), i = 1, 2, \tag{6.15}$$

where $\mathbf{B}^i$ is a matrix with state dependent values, $d\mathbf{Z}^i = \left[v^i dt, dW_2^i, dW_3^i, \cdots, dW_{P^i}^i\right]^\mathsf{T}$, and $W_p^i(t), p = 2, \cdots, P^i$, are independent Brownian motion processes. Note that P^i does not necessarily have to be equal to N^i, the number of state variables. The first element in the stochastic differential $d\mathbf{Z}^i$ is the aggregate dendritic input $v^i dt$ driving the voltage equation. The other entries in $d\mathbf{Z}^i$ are noise terms that reflect the stochastic fluctuation in the ion channels / gating variables.

Randomness is often added to BSGs by setting $\mathbf{B}^i = \mathbf{I}$, where $\mathbf{I}$ is a $N^i \times N^i$ identity matrix. The later setting can be viewed as adding subunit noise [67]. Recently, it has been suggested that a different way of adding channel noise into the BSGs may result in more accurate stochastic behavior [67, 68, 122, 143]. The SDEs in (6.15) are of general form and do not preclude them. In fact, by setting $\mathbf{B}^i$ to be a block matrix with blocks equal to be the square root of the diffusion matrix for each ion channel, the channel SDE model [68, 143] can easily be incorporated into (6.15).

Finally, we note that, under appropriate technical conditions the SDE formulation applies to BSGs with voltage-gated ion channels as well as other types of ion channels. The conditions require that the BSG model can be treated mathematically as a system of SDEs of the form (6.15) and that the latter satisfies the assumptions of Section 6.1.4.1.

6.1.4.3 Overall Encoding of the Neural Circuit Model

Taking into account the dendritic input from the feedforward DSPs and feedback DSPs, the encoding by the neural circuit model under the two noise sources is given by two systems of SDEs. With the Brownian motion W_1^i modeling the DSP white noise, the encoding of neuron $i, i = 1, 2$, can be expressed as

$$d\mathbf{Y}^i = \mathbf{f}^i(\mathbf{Y}^i, I^i)dt + \mathbf{B}^i(\mathbf{Y}^i)d\mathbf{Z}^i(t), \tag{6.16}$$

where

$$d\mathbf{Z}^i = \begin{bmatrix} v^i dt + dW_1^i \\ dW_2^i \\ \vdots \\ dW_{P^i}^i \end{bmatrix},$$

with $v^i(t)$ given by equation (6.12).

Note that in the system of equations (6.16) the two output spikes trains $(t_k^i), k \in \mathbb{Z}, i = 1, 2$, are the observables. Due to the intrinsic noise sources in the DSPs and in the BSGs, spike timing jitter may be observed from trial to trial by repeatedly applying the same stimulus to the neural circuit. This is illustrated in Figure 6.4.

An example of raster plot of the output spikes generated by the two neurons subject to 50 trials of the same stimulus is shown in Figure 6.4. We used the feedforward kernels of Example 2 and set the feedback kernels to be zero. Hodgkin-Huxley neurons in Appendix A.3.1 are used for BSGs. We set the scaling factors $\sigma_n^i = 0.01, n = 1, 2, 3, 4, i = 1, 2$. Since the initial conditions were the same for each trial, we see that initially the spikes are closer to each other across trials. As time progresses, the variability in spike times is increasing and is clearly visible.

6.2 Encoding, Decoding and Noise

In Section 6.2.1 we present the mathematical encoding formalism underlying the neural circuit in Figure 6.1. We formulate stimulus decoding as a smoothing spline optimization problem and derive an algorithm that reconstructs the encoded signal in Section 6.2.2. Finally, we analyze the effect of noise on stimulus decoding in Section 6.2.3.

6.2.1 Encoding

In this section, we formulate a rigorous stimulus encoding model based on the neural circuit shown in Figure 6.1. The input of the circuit is a signal u_1 modeling a typical sensory stimulus as described in Section 6.1.2.1. The neural circuit generates a multidimensional spike train that is assumed to be observable. We establish model equations by first de-

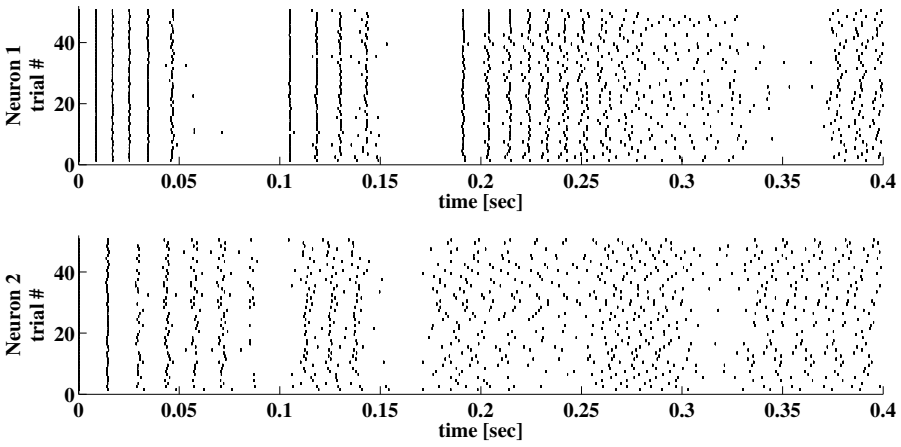


Figure 6.4: Raster plot of spikes generated by the neural circuit of Figure 6.1. The same stimulus is applied 50 times and the spike times are recorded for both neurons. The feedforward kernels employed in Example 2 were used and the feedback kernels were set to zero. The Hodgkin-Huxley neurons described in Appendix A.3.1 are used for the BSGs. Variability in spike timing is clearly visible with repeated presentation of the same stimulus.

scribing the I/O relationship (i.e., the t-transform) of a single BSG. We then provide the t-transform of the entire neural circuit model that maps the input stimulus amplitude into a multidimensional spike timing sequence.

6.2.1.1 The I/O of the BSG

In the presence of a bias current I^i and absence of external inputs, each BSG in Figure 6.1 is assumed to be periodically spiking. Provided that the inputs are small enough, and by using the PRC, the BSG dynamics of spike generation can be described in an one-dimensional phase space [95].

Definition 11. *A neuron whose spike times* $(t_k^i), k \in \mathbb{Z}$, $i = 1, 2$, *verify the system of equations*

$$\int_{t_k^i}^{t_{k+1}^i} \left[\boldsymbol{\psi}^i(s - t_k^i + \tau^i(s - t_k^i, I^i), I^i)\right]^{\mathsf{T}} \mathbf{B}^i(\mathbf{x}^i(s - t_k^i + \tau^i(s - t_k^i, I^i), I^i)) d\mathbf{Z}^i(s) = T^i(I^i) - (t_{k+1}^i - t_k^i), \tag{6.17}$$

where

$$d\tau^i(t-t_k^i, I^i) = \left[\boldsymbol{\psi}^i(t-t_k^i+\tau^i(t-t_k^i, I^i), I^i)\right]^{\mathsf{T}} \mathbf{B}^i(\mathbf{x}^i(t-t_k^i+\tau^i(t-t_k^i, I^i), I^i))d\mathbf{Z}^i(t), \quad (6.18)$$

with $\tau^i(0, I^i) = 0$ *and* $\mathbf{x}^i(t, I^i)$ *the periodic solution to* (6.13) *with bias current* I^i*, is called a Project-Integrate-and-Fire (PIF) neuron with random thresholds. In* (6.17)*,* $T^i(I^i)$ *is the period of limit cycle with bias current* I^i*.*

As its name suggests, the PIF projects a weighted version of the input embedded in noise and the ion channel noise associated with the gating variables ($\mathbf{B}^i d\mathbf{Z}^i$) onto the PRCs of the corresponding gating variables on a time interval between two consecutive spikes. Note that the integrand in (6.17) is identical to the RHS of (6.18). $\tau^i(t, I^i)$ on the LHS of (6.18) denotes the phase deviation and is driven by the perturbation on the RHS. The LHS of (6.17) represents the phase deviation measurement performed by the PIF neuron. The RHS of (6.17) provides the value of the measurement and is equal to the difference between the inter-spike interval and the period of the limit cycle.

The BSG and the PIF neuron with random thresholds are, to the first order, I/O equivalent [95]. In [95] it was also shown that a good approximation to the PIF neuron is the reduced PIF with random threshold. The functional description of the reduced PIF is obtained by setting the phase deviation in (6.17) to zero.

Definition 12. *The reduced PIF neuron with random threshold is given by the equations*

$$\sum_{n=1}^{N} \int_{t_k^i}^{t_{k+1}^i} \psi_n^i(s-t_k^i, I^i) b_{n1}^i(\mathbf{x}^i(s-t_k^i, I^i)) v^i(s) ds = T^i(I^i) - (t_{k+1}^i - t_k^i) + \varepsilon_k^i, \quad (6.19)$$

where $(\varepsilon_k^i), k \in \mathbb{Z}$*, is a sequence of independent Gaussian random variables with zero mean and variance*

$$(\mathbb{E}\left[\varepsilon_k^i\right]^2)(I^i) = \sum_{p=1}^{P^i} \int_{t_k^i}^{t_{k+1}^i} \left[\sum_{n=1}^{N^i} \psi_n^i(s-t_k^i, I^i) b_{np}^i(\mathbf{x}^i(s-t_k^i, I^i))\right]^2 ds. \quad (6.20)$$

For reasons of notational simplicity and without loss of generality, and unless otherwise stated, we shall assume here that $\mathbf{B} = \mathbf{I}$ ($N^i = P^i$). The reduced PIF (rPIF) with random threshold can now be written as

$$\int_{t_k^i}^{t_{k+1}^i} \psi_1^i(s-t_k^i, I^i) v^i(s) ds = T^i(I^i) - (t_{k+1}^i - t_k^i) + \varepsilon_k^i, \quad (6.21)$$

where $(\varepsilon_k^i), k \in \mathbb{Z}, i = 1, 2$, is a sequence of independent Gaussian random variables with zero mean and variance

$$(\mathbb{E}\left[\varepsilon_k^i\right]^2)(I^i) = \sum_{n=1}^{N^i} \int_{t_k^i}^{t_{k+1}^i} [\psi_n^i(s - t_k^i, I^i)]^2 ds. \tag{6.22}$$

The above analysis assumes that the inputs are weak and therefore the BSGs operate on a limit cycle. Stronger signals can be taken into account by considering a manifold of PRCs associated with a wide range of limit cycles [88]. By estimating the limit cycle and hence its PRC using spike times, we have the following I/O relationship for each of the BSGs.

Definition 13. *The reduced PIF neuron with conditional PRC and random threshold is given by the system of equations*

$$\int_{t_k^i}^{t_{k+1}^i} \psi_1^i(s - t_k^i, b_k^i)(v^i(s) - b_k^i + I_0^i) ds = \varepsilon_k^i, \tag{6.23}$$

where $b_k^i = [T^i]^{-1}(t_{k+1}^i - t_k^i), k \in \mathbb{Z}$, is the total estimated bias current on the inter-spike interval $[t_k^i, t_{k+1}^i]$, I_0^i is an initial bias that brings the neuron close to the spiking region in the absence of input and (by abuse of notation) $\varepsilon_k^i, k \in \mathbb{Z}, i = 1, 2$, is a sequence of independent Gaussian random variables with zero mean and variance

$$(\mathbb{E}\left[\varepsilon_k^i\right]^2)(b_k^i) = \sum_{n=1}^{N^i} \int_{t_k^i}^{t_{k+1}^i} [\psi_n^i(s - t_k^i, b_k^i)]^2 ds, \tag{6.24}$$

and $\psi_1^i(s, b_k^i)$ is the conditional PRC [88].

The conditional PRC formulation above allows us to separate BSG inputs into a constant bias current and fluctuations around it on short inter-spike time intervals. The bias current can be estimated between consecutive spikes, making the deviation from the limit cycle small in each inter-spike interval even for strong inputs. Moreover, by considering the conditional PRCs, the assumption that BSGs oscillate in the absence of input can be nearly dropped. Thus, it is not required for BSGs to always be on a limit cycle. Only when the neuron enters the limit cycle do we consider formulating the encoding using the rPIF model with conditional PRCs.

Remark 6. *Note that by parametrizing each of the PRCs with b_k^i, the variance of the error in* (6.24) *depends on the estimated PRC on each inter-spike interval. In conjunction with* (6.23)*, we see that the variability of spike times depends on the strength of the input to the BSGs.*

6.2.1.2 The t-transform of the Neural Circuit

For the set of spike indices $\mathbb{I}^i$ (see definition 5), the overall encoding by the neural circuit model can be expressed as

$$\int_{t_k^i}^{t_{k+1}^i} \psi_1^i(s-t_k^i,b_k^i)v^i(s)ds = (b_k^i - I^i)\int_{t_k^i}^{t_{k+1}^i} \psi_1^i(s-t_k^i,b_k^i)ds + \varepsilon_k^i, k \in \mathbb{I}^i, i = 1,2.$$

Substituting (6.12) into the above, we have

$$\begin{aligned}
&\int_{t_k^i}^{t_{k+1}^i} \psi_1^i(s-t_k^i,b_k^i)\int_{\mathbb{D}_1} h_1^{s|si}(s-r)u_1(r)drds \\
&+\int_{t_k^i}^{t_{k+1}^i} \psi_1^i(s-t_k^i,b_k^i)\int_{\mathbb{D}_2} h_2^{s|si}(s-r_1,s-r_2)u_1(r_1)u_1(r_2)dr_1dr_2ds \\
=&(b_k^i - I^i)\int_{t_k^i}^{t_{k+1}^i} \psi_1^i(s-t_k^i,b_k^i)ds - \sum_{l\in\mathbb{Z}}\int_{t_k^i}^{t_{k+1}^i} \psi_1^i(s-t_k^i,b_k^i)(\mathcal{P}_1^a h_1^{a|ji})(s-t_l^j)ds \\
&-\sum_{l\in\mathbb{Z}}\sum_{m\in\mathbb{Z}}\int_{t_k^i}^{t_{k+1}^i} \psi_1^i(s-t_k^i,b_k^i)(\mathcal{P}_2^a h_2^{a|ji})(s-t_l^j,s-t_m^i)ds + \varepsilon_k^i, i,j = 1,2, i\neq j.
\end{aligned} \tag{6.25}$$

We arrived at the following.

Lemma 3. *The model of encoding in Figure 6.1 is given in operator form by*

$$\mathcal{T}_{1k}^i u_1 + \mathcal{T}_{2k}^i u_2 = q_k^i + \epsilon_k^i, k \in \mathbb{I}^i, i = 1,2, \tag{6.26}$$

where $u_1 \in \mathcal{H}_1^s$, $u_2 \in \mathcal{H}_2^s, u_2(t_1,t_2) = u_1(t_1)u_1(t_2)$, and, $\mathcal{T}_{1k}^i : \mathcal{H}_1^s \to \mathbb{R}$ and $\mathcal{T}_{2k}^i : \mathcal{H}_2^s \to \mathbb{R}$ are bounded linear functionals given by

$$\mathcal{T}_{1k}^i u_1 = \int_{t_k^i}^{t_{k+1}^i} \varphi_k^i(s)\int_{\mathbb{D}_1} h_1^{s|si}(s-r)u_1(r)drds,$$

$$\mathcal{T}_{2k}^i u_2 = \int_{t_k^i}^{t_{k+1}^i} \varphi_k^i(s)\int_{\mathbb{D}_2} h_2^{s|si}(s-r_1,s-r_2)u_2(r_1,r_2)dr_1dr_2ds,$$

$$q_k^i = (b_k^i - I^i)\int_{t_k^i}^{t_{k+1}^i} \varphi_k^i(s)ds - \sum_{l\in\mathbb{Z}} \int_{t_k^i}^{t_{k+1}^i} \varphi_k^i(s)(\mathcal{P}_1^a h_1^{a|ji})(s - t_l^j)ds$$
$$- \sum_{l\in\mathbb{Z}}\sum_{m\in\mathbb{Z}} \int_{t_k^i}^{t_{k+1}^i} \varphi_k^i(s)(\mathcal{P}_2^a h_2^{a|ji})(s - t_l^j, s - t_m^i)ds,$$
$$\varphi_k^i(t) = \frac{\psi_1^i(t - t_k^i, b_k^i)}{\left(\mathbb{E}\left[\varepsilon_k^i\right]^2\right)^{\frac{1}{2}}}$$

and $\epsilon_k^i, k \in \mathbb{I}^i, i = 1, 2$*, are independent random variables with normal distribution* $\mathcal{N}(0, 1)$ *and* $j = 1, 2, j \neq i$*. Equation* (6.26) *is called the t-transform [106] of the neural circuit in Figure 6.1.*

Remark 7. *The t-transform describes the mapping of the input stimulus* u_1 *into the spike timing sequence* $(t_k^i), k \in \mathbb{I}^i, i = 1, 2$*. Thus, the t-transform shows how the amplitude information of the input signal is related to or transformed into the time information contained in the sequence of output spikes generated by the neural circuit.*

We provide here further intuition behind the equations (6.26). By the Riesz representation theorem [10], there exists functions $\phi_{1k}^i \in \mathcal{H}_1^s$ such that

$$\mathcal{T}_{1k}^i u_1 = \langle u_1, \phi_{1k}^i \rangle_{\mathcal{H}_1^s}, \text{ for all } u_1 \in \mathcal{H}_1^s,$$

and $\phi_{2k}^i \in \mathcal{H}_2^s$ such that

$$\mathcal{T}_{2k}^i u_2 = \langle u_2, \phi_{2k}^i \rangle_{\mathcal{H}_2^s}, \text{ for all } u_2 \in \mathcal{H}_2^s.$$

Therefore, (6.26) can be rewritten in inner product form:

$$\langle u_1, \phi_{1k}^i \rangle_{\mathcal{H}_1^s} + \langle u_2, \phi_{2k}^i \rangle_{\mathcal{H}_2^s} = q_k^i + \varepsilon_k^i. \tag{6.27}$$

Recall that inner products are projections that are typically interpreted as measurements. In the equations (6.27) above, the signals u_1 and u_2 are projected onto the sampling functions ϕ_{1k}^i and ϕ_{2k}^i, respectively. We also note that traditional amplitude sampling of a bandlimited signal u_1 at times $(t_n), n \in \mathbb{Z}$, can be expressed as

$$\langle u_1(\cdot), \text{sinc}(t_n - \cdot) \rangle_{\mathbb{L}^2(\mathbb{R})} = u_1(t_n),$$

where $\text{sinc}(t) = \frac{\sin(\Omega_t^s t)}{\pi t}$ is the impulse response of the ideal low pass filter bandlimited to Ω_t^s or in other words, the kernel of the RKHS of finite-energy band-limited functions [97]. Thus, the neural encoding model described by the equations (6.27) can be interpreted as generalized sampling with noisy measurements with sampling functions ϕ_{1k}^i and ϕ_{2k}^i.

The formulation of the encoding model can easily be extended to the case when M neural circuits encode a stimulus in parallel. This is shown schematically in Figure 6.2. A left superscript was added in the figure to each of the components to indicate the circuit number.

6.2.2 Decoding

In the previous section, we showed that the encoding of a signal u_1 by the neural circuit model with feedforward and feedback DSPs and BSGs can be characterized by the set of t-transform equations (6.26). We noticed that the equations (6.26) are nonlinear in u_1 due to the second order Volterra term. However, by reinterpreting the second order term as linear functionals $\mathcal{T}_{2k}^i$ on the higher dimensional tensor space $\mathcal{H}_2^s$, (6.26) implies that the measurements taken by each of the neurons are the sum of linear measurements in two different vector spaces (see also equations (6.27)).

In this section we investigate the decoding of signals encoded with the neural circuit in Figure 6.1. The purpose of decoding is to recover from the set of spike times the original signals, $u_1(t)$ and $u_2(t_1, t_2)$, that respectively belong to the two different vector spaces $\mathcal{H}_1^s$ and $\mathcal{H}_2^s$. We formulate the decoding problem as the joint smoothing spline problem

$$(\hat{u}_1, \hat{u}_2) = \underset{u_1 \in \mathcal{H}_1^s, u_2 \in \mathcal{H}_2^s}{\text{argmin}} \{\lambda_1 \|u_1\|_{\mathcal{H}_1^s}^2 + \lambda_2 \|u_2\|_{\mathcal{H}_2^s}^2 + \sum_{i=1}^{2} \sum_{k \in \mathbb{I}^i} (\mathcal{T}_{1k}^i u_1 + \mathcal{T}_{2k}^i u_2 - q_k^i)^2\}, \tag{6.28}$$

where $\mathbb{I}^i$ is the index set of spikes generated by BSG $i = 1, 2$.

Theorem 5. *The solution to* (6.28) *is of the form*

$$\begin{aligned} \hat{u}_1(t) &= \sum_{i=1}^{2} \sum_{k \in \mathbb{I}^i} c_k^i \phi_{1k}^i(t) \\ \hat{u}_2(t_1, t_2) &= \sum_{i=1}^{2} \sum_{k \in \mathbb{I}^i} c_k^i \phi_{2k}^i(t_1, t_2) \end{aligned}, \tag{6.29}$$

where $\phi^i_{1k}(t) = \mathcal{T}^i_{1k} K^s_{1|t}$ *and* $\phi^i_{2k}(t_1, t_2) = \mathcal{T}^i_{2k} K^s_{2|t_1,t_2}, k \in \mathbb{I}^i, i = 1, 2,$
$\mathbf{c} = \left[c^1_1, \cdots, c^1_{|\mathbb{I}^1|}, c^2_1, \cdots, c^2_{|\mathbb{I}^2|}\right]^{\mathsf{T}}$ *is the solution of the system of linear equations*

$$\left((\mathbf{\Phi}_1 + \mathbf{\Phi}_2)^2 + \lambda_1 \mathbf{\Phi}_1 + \lambda_2 \mathbf{\Phi}_2\right) \mathbf{c} = (\mathbf{\Phi}_1 + \mathbf{\Phi}_2)\mathbf{q}, \tag{6.30}$$

where $\mathbf{q} = \left[q^1_1, \cdots, q^1_{|\mathbb{I}^1|}, q^2_1, \cdots, q^2_{|\mathbb{I}^2|}\right]^{\mathsf{T}}$, *and*

$$\mathbf{\Phi}_i = \begin{bmatrix} \mathbf{\Phi}_i^{11} & \mathbf{\Phi}_i^{12} \\ \mathbf{\Phi}_i^{21} & \mathbf{\Phi}_i^{22} \end{bmatrix}, i = 1, 2,$$

and

$$[\mathbf{\Phi}_i^{mn}]_{kl} = \langle \phi^m_{ik}, \phi^n_{il} \rangle, \text{ for all } k \in \mathbb{I}^m, l \in \mathbb{I}^n, m, n = 1, 2, i = 1, 2.$$

Proof: Proof of the theorem follows the Representer Theorem [10] and is given in detail in Appendix B.5.

Remark 8. *When* $\lambda_1 = \lambda_2$, *the solution* $\mathbf{c}$ *amounts to*

$$\mathbf{c} = (\mathbf{\Phi}_1 + \mathbf{\Phi}_2 + \lambda_1 \mathbf{I})^{-1} \mathbf{q},$$

where $\mathbf{I}$ *is an identity matrix of appropriate dimensions.*

Remark 9. *Although* (6.29) *solves* (6.28), *in practice a minimum number of spikes is needed to obtain a meaningful estimate of the original signal. A minimum bound for the number of measurements/spikes can be derived in the noiseless case. Clearly, the bound has to be larger than the dimension of the space. This may require the signal to be encoded by a circuit with a larger number of neurons than the two shown in Figure 6.1 [105]. A number of such neural circuits in parallel can be used to encode input stimuli as shown in Figure 6.2. Theorem 5 can be easily extended to solving the smoothing spline problem*

$$(\hat{u}_1, \hat{u}_2) = \underset{u_1 \in \mathcal{H}^s_1, u_2 \in \mathcal{H}^s_2}{\operatorname{argmin}} \{\lambda_1 \|u_1\|^2_{\mathcal{H}^s_1} + \lambda_2 \|u_2\|^2_{\mathcal{H}^s_2} + \sum_{m=1}^{M} \sum_{i=1}^{2} \sum_{k \in {}^m\mathbb{I}^i} ({}^m\mathcal{T}^i_{1k} u_1 + {}^m\mathcal{T}^i_{2k} u_2 - {}^m q^i_k)^2\},$$

where $m = 1, 2, ..., M$, *denotes the circuits number in Figure 6.2. In addition, if the circuits consist of only first order feedforward kernels, then only* $u_1(t)$ *can be reconstructed.*

Similarly, if the circuits are comprised of only the second order feedforward kernels, then $u_2(t_1, t_2)$ *can be reconstructed but not* $u_1(t)$.

Remark 10. *Since* $u_2(t_1, t_2) = u_1(t_1)u_1(t_2) = u_2(t_2, t_1)$, u_2 *belongs to a subspace of* $\mathcal{H}_2^s$ *whose elements are symmetric functions. We also note that since the second order feedforward kernels are symmetric, the sampling functions* $(\phi_{2k}^i(t_1, t_2)), k \in \mathbb{I}^i, i = 1, 2$, *also belong to the same subspace. Therefore, if the sampling functions span the subspace of symmetric functions in* $\mathcal{H}_2^s$, u_2 *can readily be reconstructed with only* $(L_t^s + 1)(2L_t^s + 1)$ *measurements/spikes, rather than* $(2L_t^s + 1)^2$, *the dimension of* $\mathcal{H}_2^s$.

Remark 11. *The reconstruction of* $u_2(t_1, t_2)$ *on* $\mathbb{D}_2$ *strongly depends on the support (in practice the finite memory) of the kernels* $h_2^{s|si}, i = 1, 2$ *(see also Figure 6.3(c)). In the reconstruction example of the Figure 6.5, we show that* $\hat{u}_2$ *approximates* u_2 *well in the restricted domain where* $h_2^{s|si}$ *is non-zero. Outside this restricted domain,* $h_2^{s|si}$ *vanishes and* u_2 *is not well recovered as suggested by the large error in the Figure 6.5(e).*

Example 3. *We show an example here under noiseless condition. This can be used as a baseline for example in Section 3.3.2 where internal noise sources are present in the neural circuits.*

We consider encoding a 0.4 *[s] signal bandlimited to* 10 *[Hz] using neural circuits described in Figure 6.2, with* $M = 4$. *That is, a total of* 8 *neurons are used for encoding. The order of the input space is* $L = 4$.

We choose the following feedforward and feedback kernels for neural circuit 1:

$$
\begin{aligned}
{}^1h_1^{s|s1}(t) &= 400\left(\exp(-100t)\,\frac{(100t)^3}{3!} - \exp(-100t)\,\frac{(100t)^5}{5!}\right),\\
{}^1h_2^{s|s1}(t_1, t_2) &= 16(g^1(t_1)g^1(t_2) + g^2(t_1)g^2(t_2)),\\
{}^1h_1^{s|s2}(t) &= 0,\\
{}^1h_2^{s|s2}(t_1, t_2) &= 16(g^1(t_1)g^1(t_2) - g^2(t_1)g^2(t_2)),\\
{}^1h_1^{a|21}(t) &= 20\exp\left(-200(t - 10^{-3})\right)\frac{(200(t - 10^{-3}))^3}{3!},
\end{aligned}
$$

$$ {}^{1}h_1^{a|12}(t) = 10 \exp\left(-200(t-10^{-3})\right) \frac{(200(t-10^{-3}))^3}{3!}, $$

The DSP kernels for the rest of three neural circuits uses variations of the above, e.g., different scales (dilations), weights and signs.

We choose a Hodgkin-Huxley neuron in this example. The bias current of all the neurons is set to $10\mu A/cm^2$*. We used a simple forward Euler scheme [60] in simulations the Hodgkin-Huxley neuron with integration time step* 10^{-6}*[s]. The time step is small enough to guarantee the stability and precision of numerical integration. We did not use higher order methods, for example, the staggered Euler scheme [75] since the Euler scheme may be more directly comparable to Euler-Maruyama scheme we employed in the stochastic case.*

A stimulus is constructed using equation (1) with complex coefficients with real and imaginary parts randomly chosen from a standard normal distribution. A total of 155 *spikes are generated from all* 8 *neurons. Among those* 93 *are valid measurements. We note that since the t-transform is approximate even under noiseless condition, introducing a smoothing parameters is still necessary. We set* $\lambda_1 = \lambda_2 = \lambda$ *and used leave-out-one cross validation to find the optimal* λ*. We leaved out the valid measurements from one neuron each time in the cross validation. We determined the optimal* $\lambda = 8.25 \times 10^{-8}$*. The reconstruction of* $u_1(t)$ *is shown in Figure 6.5(a). Signal-to-Noise Ratio (SNR) is* 25.29 *[dB]. The error of the reconstruction of* $u_2(t_1, t_2)$ *is shown in Figure 6.5(e). As suggested, only the part that is sampled by the second order feedforward kernels is recovered. This can be seen from the overlaid kernel on the reconstruction. In particular,* $u_1^2(t)$ *is recovered as* $\hat{u}_2(t,t)$ *and the SNR of this recovery is* 24.85 *[dB] (see Figure 6.5(b)). By comparison, the SNR of reconstructions without smoothing, i.e.,* $\lambda = 0$*, are* 19.14 *[dB] for* $u_1(t)$ *and* 8.70 *[dB] for* $u_1^2(t)$*.*

Note that in this example, the space $\mathcal{H}_1^s$ *is of dimension* $2L_t^s + 1 = 9$*, and* $\mathcal{H}_2^s$ *is of dimension* $(2L_t^s + 1)^2 = 81$*. However, since the second order feedforward kernel are all symmetric, they generate a subspace of symmetric functions when sampling* $u_2(t_1, t_2)$*. This subspace is of dimension* $(L_t^s + 1)(2L_t^s + 1) = 45$*. In addition, each neuron can generate upto* $2 \cdot 2L_t^s + 1$ *linearly independent sampling functions. Therefore, the minimum number of neurons required for faithfully representing the input stimuli is* $(45 + 9)/(2 \cdot 2L_t^s + 1) = 4$*.*

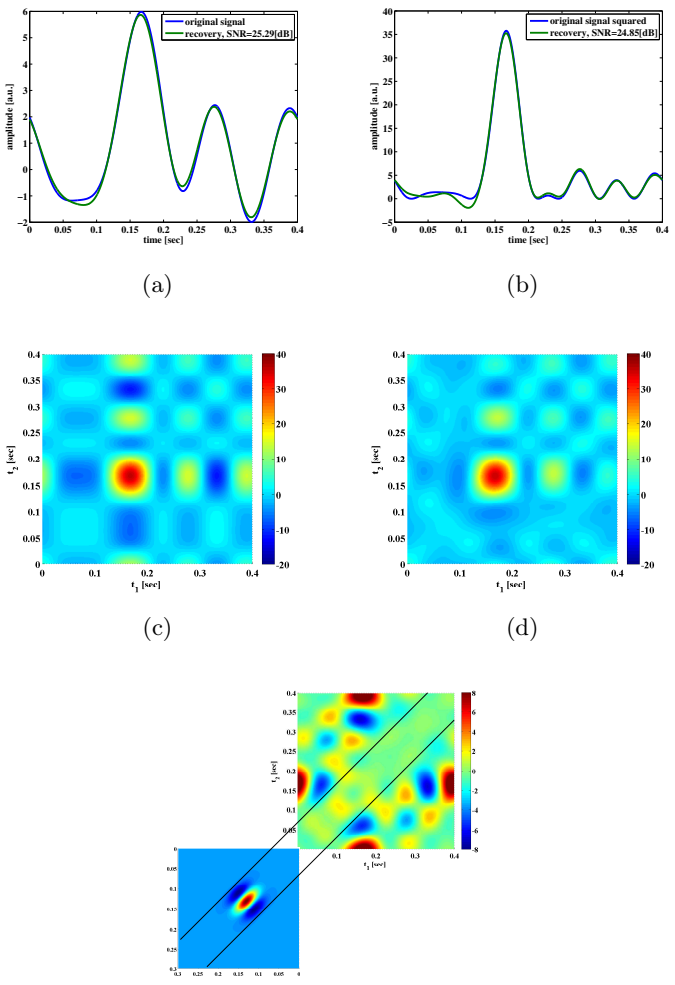


Figure 6.5: Examples of decoding under noiseless condition. (a) Original signal u_1 (blue) and its reconstruction (green). (b) Original $u_1^2(t)$ (blue) and its reconstruction (green). (c) Original $u_2(t_1, t_2) = u_1(t_1)u_1(t_2)$. (d) Reconstruction $\hat{u}_2(t_1, t_2)$. (e) Error between original $u_2(t_1, t_2) = u_1(t_1)u_1(t_2)$ and its reconstruction (top). When evaluating the second order feedforward DSP output, u_2 (in (C)) is multiplied by nonzero values of $h_2^{s|si}$ (bottom) only in the domain between the black lines. u_2 in this domain is well reconstructed, whereas it is poorly reconstructed outside of this domain.

6.2.3 Effect of Noise on Stimulus Decoding

In this section, we investigate the effect of noise sources (i) on spike timing of the reduced PIF neuron, and (ii) on the decoding of stimuli encoded with a neural circuit. We will also present the effect of an alternative noise source model on both spike timing and stimulus decoding.

6.2.3.1 Effect of Noise on Measurement and Spike Timing Errors of the Reduced PIF Neuron

As suggested by (6.22), the variance of the measurement error of the reduced PIF neuron is directly related to the PRC of the associated limit cycle. We first characterize the variance of the measurement error due to each individual noise source parametrized by the bias current I^i. We then evaluate the spike timing variance between the spike trains generated by the Hodgkin-Huxley neuron and the reduced PIF neuron again as a function of the bias current I^i. We start with a brief description of the key elements of Hodgkin-Huxley neuron and the PIF neuron.

We consider the stochastic Hodgkin-Huxley equations

$$d\mathbf{Y}^i = \mathbf{f}^i(\mathbf{Y}^i, I^i)dt + d\mathbf{Z}^i(t), \tag{6.31}$$

where $\mathbf{f}^i$ is defined as in Appendix A.3.1 with additional normalization such that the unit of time is in seconds instead of milliseconds and the unit of voltage is in Volts instead of milivolts as conventionally used.

The variance of the measurement error of the reduced PIF neuron due to each Brownian motion process $W_n^i, n = 1, \cdots, 4$ is given by (see also equation (6.22))

$$(\mathbb{E}\left[\varepsilon_{kn}^i\right]^2)(I^i) = (\sigma_n^i)^2 \int_{t_k^i}^{t_{k+1}^i} [\psi_n^i(s - t_k^i, I^i)]^2 ds. \tag{6.32}$$

We show in Figure 6.6(a) the variance of the measurement error in (6.32) associated with each source of noise of the reduced PIF neuron for the unitary noise levels $\sigma_n^i = 1, n = 1, 2, 3, 4$. The variances given by (6.32) are plotted as a function of the bias current I^i. Clearly, the noise arising in dendritic stimulus processing (W_1^i) induces the largest error, and together

with noise in the potassium channels (W_2^i), these errors are about two magnitudes larger in variance than those induced by the noise sources in the sodium channels (W_3^i, W_4^i).

The above analysis is based on the analytical derivation of the measurement error in (6.32) for the rPIF neurons. The measurement error is closely related, however, to the spike timing variation of the BSGs subject to noise sources. A variance of 10^{-6} in Figure 6.6(a) corresponds to a standard deviation of 1 [msec] in spike timing. In practice the error between the spike times of the Hodgkin-Huxley neuron and the reduced PIF neuron can be directly evaluated.

In order to do so, we randomly generated a weak bandlimited dendritic input. All evaluations were based on encoding a signal with the Hodgkin-Huxley neuron model described above with internal noise sources and bias current I^i. The spike times (t_k^i) of the Hodgkin-Huxley neuron were recorded. Starting from each spike time t_k^i, we encoded the appropriate portion of the signal by the reduced PIF neuron until a spike ${}^r t_{k+1}^i$ was generated. The difference between ${}^r t_{k+1}^i$ and t_{k+1}^i is the error in approximating the encoding using the reduced PIF formulation. This process was repeated for each I^i. We computed the variance of the errors based on some $3,000$ to $5,000$ spikes generated in encoding the input.

In Figure 6.6(b), the variance of the spike timing error ${}^r t_{k+1}^i - t_{k+1}^i$ for $\sigma_n = 0, n = 1, 2, 3, 4$, is shown. Since the reduced PIF is an approximation (even under noiseless conditions) and, although small, the error is nonzero. From Figure 6.6(b), the variance of the spike timing error is on the order of 10^{-9}. We shall evaluate the spike timing error variance of the intrinsic noise sources in a range much larger than 10^{-9}.

We also tested to what extent each individual source of noise contributes to the variance of spike timing as suggested by the theoretical analysis depicted in Figure 6.6(a). Indeed, the error variance obtained through simulations in Figure 6.6(c) follows the basic pattern shown in Figure 6.6(a). Figure 6.6(c) was obtained by setting one of the σ_n's to a nonzero value and the rest to 0 (the nonzero values were $\sigma_1 = \sigma_2 = 0.01$, $\sigma_3 = \sigma_4 = 0.1$). Each nonzero value was picked to be large enough so that the error variance in the absence of noise (Figure 6.6(b)) becomes negligible, and at the same time, it was small enough such that the states of the neurons did not substantially deviate from the limit cycle. To compare the with the ones in Figure 6.6(a) we normalized the error variance obtained in simulations

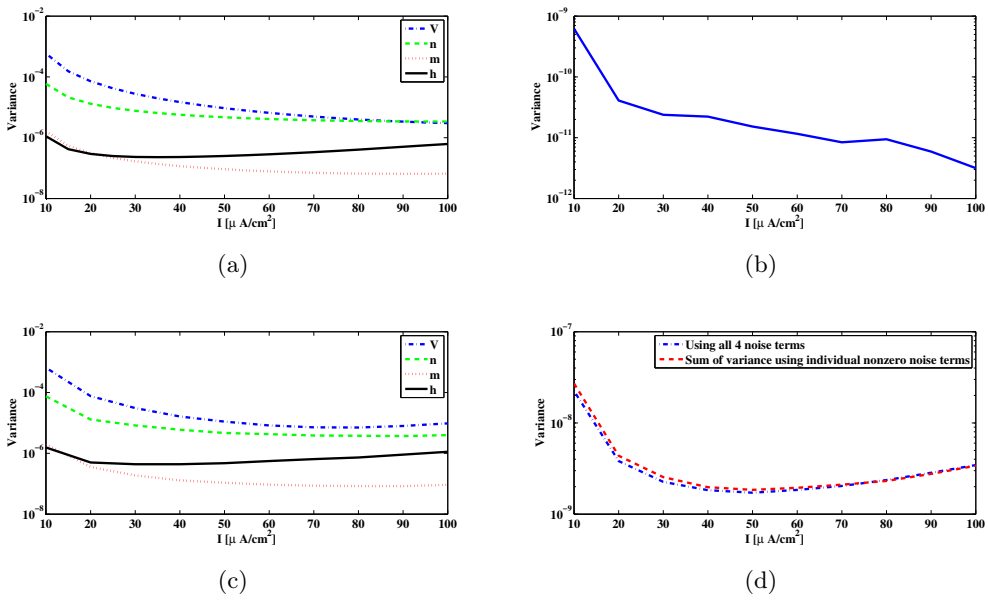


Figure 6.6: Variance of the measurement and spike timing errors. (a) Error measurement variances computed from the PRCs of the Hodgkin-Huxley neuron (equation (6.32)). Each individual variance is parametrized by the bias current I^i. (b) Error variance between spike times generated by the noiseless Hodgkin-Huxley neuron and its reduced PIF counterpart. (c) The spike timing error variance due to each source of noise, obtained from simulations of the Hodgkin-Huxley neuron follow the pattern of the theoretically derived measurement error shown in (A). The spike timing error variances are obtained by setting, at each time, one of the σ_n's to a nonzero value and the rest to zero. The spikes generated by the Hodgkin-Huxley neuron are compared with the spikes generated by its reduced PIF counterpart. The variance of the differences between two spike times are normalized by the nonzero σ_n mentioned before. (D) The spike timing variance due to the simultaneous presence of multiple noise sources approximates the sum of spike timing variances due to individual noise sources. Blue curve shows the spike timing variance obtained by simulating Hodgkin-Huxley equations using nonzero values for all $\sigma_n, n = 1, 2, 3, 4$. Red curve shows the sum of spike timing variances obtained in (C) with proper scaling.

by σ_n.

Next, we tested whether the variance of spike timing due to presence of multiple noise sources is truly the summation of error variances due to individual noise sources. We simulated the Hodgkin-Huxley equations with $\sigma_1 = \sigma_2 = 0.005$, $\sigma_3 = \sigma_4 = 0.05$. The total spike timing error variance shown in Figure 6.6(d) (blue curve) is very close to the sum of error variances in Figure 6.6(c) with proper scaling (red curve in Figure 6.6(d)).

As suggested by the above analysis, the reduced PIF neuron with random thresholds largely captures the encoding of stimuli by BSGs subject to intrinsic noise sources.

6.2.3.2 Effect of Noise on Stimulus Decoding

In order to quantitatively explore how noise impacts signal decoding, we recovered from spikes the signal encoded by the noisy neural circuit of Figure 6.2. We started with the base-level noise-less case described in example 3 and proceeded to introduce individual noise terms with a range of scaling factors. For example, we set $\sigma_2^i = \sigma_3^i = \sigma_4^i = 0$ and varied σ_1^i. We also tested the case when $10\sigma_1^i = 10\sigma_2^i = \sigma_3^i = \sigma_4^i$ for the aggregated effect on stimulus recovery. We choose to use σ_3^i and σ_4^i 10 times larger than σ_1^i and σ_2^i so that each noise source introduced a similar error.

In all simulations, the Euler-Maruyama scheme [90] was used for the numerical integration of the SDEs. We performed 20 encoding and decoding experiments. Each time, the input stimulus was generated by randomly picking from a Gaussian distribution the real and imaginary parts of the coefficients u_l in (6.1). We further constrained the stimuli to be real-valued. (An example is given in Figure 6.5.) For each noise level, the input signal was encoded/decoded. The mean Signal-to-Noise Ratio (SNR) across 20 experiments is reported for each noise level. The SNR for the reconstruction of u_1 was computed as

$$\mathrm{SNR} = 10 \log_{10} \left[\frac{\|u_1\|^2}{\|u_1 - \hat{u}_1\|^2} \right], \tag{6.33}$$

where u_1 is the original signal and $\hat{u}_1$ is its reconstruction. Note that the spike time occurrences generated for the same signal are different for each noise level. Since the sampling functions are spike time dependent, the number of measurements/spikes may not be the same for each noise level. Moreover, at times, the sampling functions may not fully span

the stimulus space. To reduce the uncertainty caused by the stimulus dependent sampling we averaged our SNR data over 20 different signals.

Figure 6.7(a) shows the SNR of the reconstruction of signal $u_1(t)$ against different noise strength. Figure 6.7(b) shows the SNR of the reconstruction of signal $u_1^2(t) = u_2(t,t)$. The reconstruction SNR in Figure 6.7(a) largely matches the inverse ordering of noise strength of each of the individual noise sources shown in Figure 6.6(a). DSP noise sources degrade the reconstruction performance most strongly while noise sources associated with gating variables m and h have a much smaller effect for the same variance level. Since the variance of measurement error is the sum of error variance in each variable, the case when $10\sigma_1 = 10\sigma_2 = \sigma_3 = \sigma_4 = \sigma$ exhibits the lowest performance.

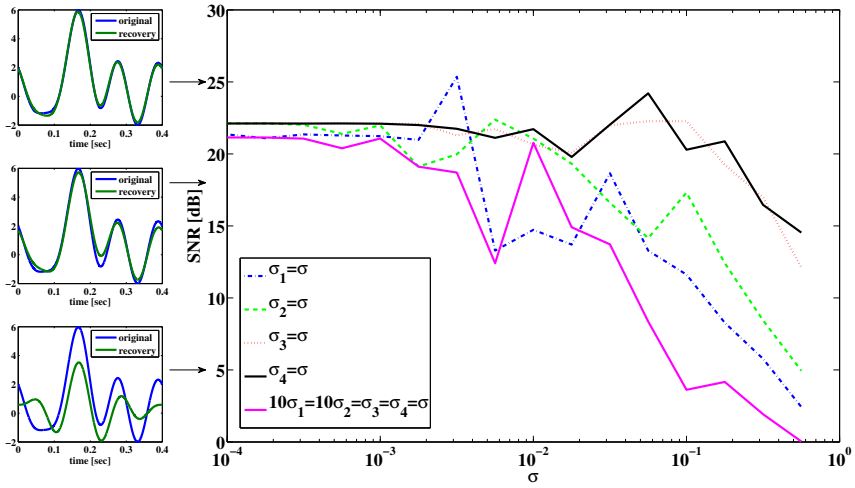
6.2.3.3 Effect of an Alternative Noise Model on Spike Timing and Ttimulus Decoding

Biologically, the effect of channel noise on the operation of the BSGs is due to the ON-OFF activity of a finite number of ion channels. The Hodgkin-Huxley equations and the noise terms used in Section 6.2.3.2 correctly capture the dynamics in the limit of infinitely many channels. Recent research, however, suggests that the model equations may not correctly model the ion current fluctuations for a finite number of channels [67].

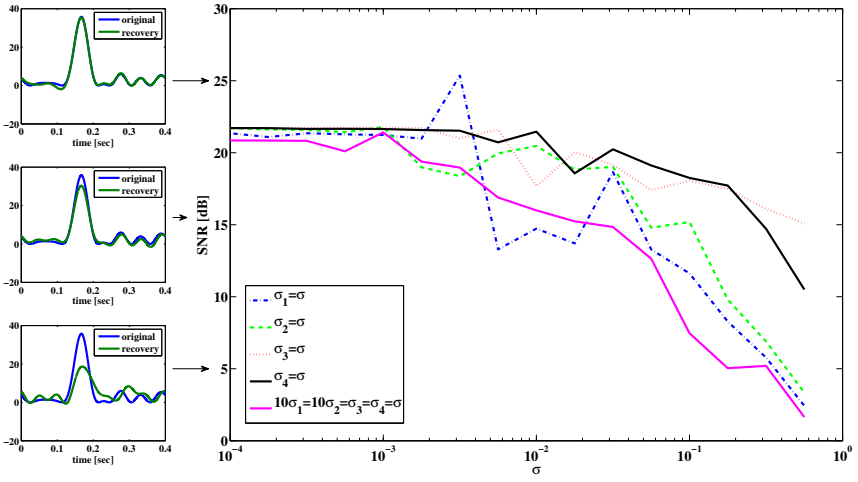
We consider here an alternative stochastic formulation of the Hodgkin-Huxley model that more precisely captures the ion channel kinetics. By using a finite number of ion channels the strength of noise amplitude becomes directly related to the actual number of ion channels. Therefore, the free variables are only the number of potassium and sodium channels that are both biologically meaningful. The successful use of an alternative noise model as described below also suggests that our analysis can be applied to a wide range of stochastic formulations of BSGs based on SDEs.

We shall construct here stochastic ion channels using conductance noise rather than subunit noise as in the previous sections [67, 68]. This stochastic Hodgkin-Huxley system is simulated using a diffusion approximation following [143]. The system of SDEs can be expressed as

$$d\mathbf{Y}^i = \mathbf{f}^i(\mathbf{Y}^i, I^i)dt + \mathbf{B}^i(\mathbf{Y}^i)d\mathbf{Z}^i(t),$$



(a)



(b)

Figure 6.7: SNR reconstruction error of encoded signals with a total of $M = 2$ circuits (4 neurons). Color legend: (Blue) $\sigma_1^i = \sigma$, $\sigma_2^i = \sigma_3^i = \sigma_4^i = 0$. (Green) $\sigma_2^i = \sigma$, $\sigma_1^i = \sigma_3^i = \sigma_4^i = 0$. (Red) $\sigma_3^i = \sigma$, $\sigma_1^i = \sigma_2^i = \sigma_4^i = 0$. (Black) $\sigma_4^i = \sigma$, $\sigma_1^i = \sigma_2^i = \sigma_3^i = 0$. (Magenta) $10\sigma_1^i = 10\sigma_2^i = \sigma_3^i = \sigma_4^i = \sigma$. In-sets (on the left) are typical reconstructions that yield corresponding SNR indicated by arrows. The top left in (A) shows an example of reconstruction (green) whose SNR is 25 [dB] when compared to the original signal (blue). (A) SNR of reconstruction of $u_1(t)$. (B) SNR of reconstruction of $u_1^2(t) = u_2(t,t)$.

where $\mathbf{Y}^i$ has 14 state variables and the full system can be found in Appendix A.3.2. Here $i = 1$ for simplicity.

The variance of the measurement error is now given by (6.20). We can decompose the variance into three terms as

$$\mathbb{E}\left[\varepsilon_k^i\right]^2 = \mathbb{E}\left[\varepsilon_{kV}^i\right]^2 + \mathbb{E}\left[\varepsilon_{kK}^i\right]^2 + \mathbb{E}\left[\varepsilon_{kNa}^i\right]^2,$$

where $\varepsilon_{kV}^i, \varepsilon_{kK}^i, \varepsilon_{kNa}^i$ are measurement errors associated with the noise in the DSP, in potassium channels and in sodium channels, respectively.

As ε_{kV}^i is quantitatively the same as that in Section 6.2.3.2, we focus here on ε_{kK}^i and ε_{kNa}^i. The variance of the errors can be respectively expressed as

$$(\mathbb{E}\left[\varepsilon_{kK}^i\right]^2)(I^i) = \sum_{p=2}^{5} \int_{t_k^i}^{t_{k+1}^i} \left[\sum_{n=2}^{6} \psi_n^i(s - t_k^i, I^i) b_{np}^i\left(\mathbf{x}^i(s - t_k^i, I^i)\right)\right]^2 ds,$$

and

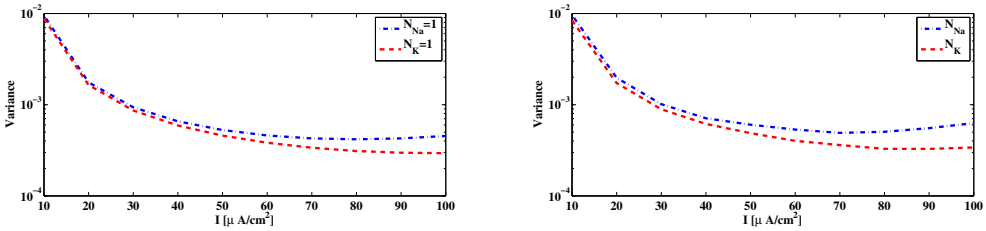
$$(\mathbb{E}\left[\varepsilon_{kNa}^i\right]^2)(I^i) = \sum_{p=6}^{15} \int_{t_k^i}^{t_{k+1}^i} \left[\sum_{n=7}^{14} \psi_n^i(s - t_k^i, I^i) b_{np}^i\left(\mathbf{x}^i(s - t_k^i, I^i)\right)\right]^2 ds.$$

Note that $b_{np}, n = 1, \cdots, 14, p = 2, 3, \cdots, 15$, are functions that dependent on either the number of potassium channels N_{Na} or the number of sodium channels N_K, and the states of the neuron.

We first evaluate $(\mathbb{E}\left[\varepsilon_{kNa}^i\right]^2)(I^i)$ using the PRCs. The PRCs are obtained by letting $N_{Na} = N_K = \infty$ and thereby making the system deterministic. Since the measurement error variance for fixed I^i is proportional to $(N_{Na})^{-1}$, it is shown in Figure 6.8 as a function of the bias current I^i for $N_{Na} = 1$. Similarly, the variance of the measurement error $(\mathbb{E}\left[\varepsilon_{kK}^i\right]^2)(I^i)$ for $N_K = 1$ is shown in Figure 6.8, and it is proportional to $(N_K)^{-1}$ for a fixed I^i. We notice that, when the number of channels is the same, the measurement error due to the sodium channels is of the same order of magnitude with the measurement error due to the potassium channels. However, the number of sodium channels is typically 3 to 4 times larger than the number of potassium channels. Therefore, in contrast to the previous case, the error induced by sodium channels shall be larger than that induced by potassium channels.

We also analyzed in simulations the difference between spike times generated by the alternative stochastic formulation of the Hodgkin-Huxley equations and those generated by

the corresponding reduced PIF neuron. We used in simulation $N_{Na} = 5 \times 10^4, N_K = \infty$, to obtain the variance $(\mathbb{E}\left[\varepsilon^i_{kNa}\right]^2)(I^i)$ and scaled it by N_{Na} to compare it with Figure 6.8. Similarly, we used $N_K = 5 \times 10^4, N_{Na} = \infty$, to obtain the variance $\mathbb{E}\left[\varepsilon^i_{kK}\right]^2(I^i)$. The spike timing variances of error across different I^i are shown in Figure 6.8(b). The pattern of similarity between variances in 6.8 and in Figure 6.8(b) suggest that the reduced PIF with random threshold associated with this formulation of Hodgkin-Huxley equations is highly effective in capturing the encoding under internal noise sources.

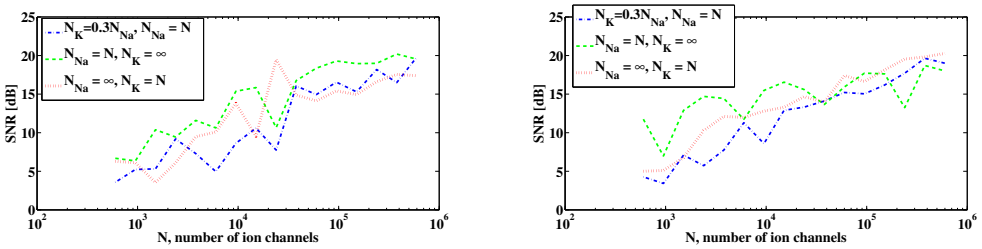


(a) (b)

Figure 6.8: The variance of the measurement and spike timing error associated with the sodium channels (blue) and the potassium channels (red) of the Hodgkin-Huxley equations with alternative noise sources parametrized by the bias current I. (a) The variance of the measurement error computed from PRCs of Hodgkin-Huxley equations, with $N_{Na} = 1$ and $N_K = 1$. Actual variance with different number of ion channels is inversely proportional to N_{Na} and N_K, respectively. (b) Spike timing variance obtained in simulations by comparing the spike times generated by the Hodgkin-Huxley neuron with channel noise and the spike times generated by its reduced PIF counterpart. Blue curve is obtained by using $N_{Na} = 5 \times 10^4, N_K = \infty$, and normalized to 1 sodium channel. Red curve is obtained by using $N_K = 5 \times 10^4, N_{Na} = \infty$, and normalized to 1 potassium channel.

We now show how ion channel noise sources affect the decoding of the input signal. We varied the number of sodium channels N_{Na} and fixed the number of potassium channels to be $N_K = 0.3N_{Na}$, a ratio typically used for Hodgkin-Huxley neurons with the alternative noise source model. By decoding the input stimulus we show how increasing the number of ion channels improves the faithfulness of signal representation. The SNR of the reconstruction of $u_1(t)$ and $u_1^2(t)$ are depicted in Figure 6.9. SNR goes down to about 4 [dB] when 600

sodium channels and 180 potassium channels are used. This corresponds to a membrane area of about $10\mu m^2$ with a density of $60/(\mu m^2)$ in sodium channels and $18/(\mu m^2)$ in potassium channels [68]. We also tested the reconstruction for the case when one type of ion channels is infinitely large, *i.e.*, deterministic, while varying the number of ion channels of the other type. The result is also shown in Figure 6.9. The noise from the dendritic tree shall have similar effect on the representation since the voltage equation is the same as in Section 6.2.3.2.



(a) (b)

Figure 6.9: SNR of reconstruction of the signals. (a) SNR of $u_1(t)$. (b) SNR of $u_1^2(t) = u_2(t,t)$. (Blue) $N_{Na} = N$, $N_K = 0.3N_{Na}$. (Green) $N_{Na} = N$, $N_K = \infty$. (Red) $N_{Na} = \infty$, $N_K = N$.

6.3 Functional Identification and Noise

In Section 6.3.1 we show how to functionally identify the feedforward and feedback DSPs of the circuit in Figure 6.1 under noisy conditions. We assume here that the BSGs have already been identified using a methodology such as the one developed in [103]. In Section 6.3.2 we discuss the effect of noise parameters on the quality of DSP identification.

6.3.1 Functional Identification

In the decoding problem analyzed in Section 6.2.2, we reconstructed unknown input stimuli by assuming that the neural circuit in Figure 6.1 is known and the spike trains are observable. In contrast, the objective of the functional identification problem investigated in this section is to estimate the unknown circuit parameters of the feedforward and feedback DSPs from

I/O data. The I/O data is obtained by stimulating the circuit with controllable inputs and by measuring the time occurrences of the output spikes. This basic methodology has been a standard practice in neurophysiology for inferring the function of sensory systems [79]. We assume here that either the BSGs are known in functional form or the family of PRCs associated with the BSGs have already been identified [103].

Although decoding and functional identification are seemingly two different problems, they are closely related. By exploiting the commutative property of linear operators, we can rearrange the diagram of the neural circuit model of Figure 6.1 into the form shown in Figure 6.10. We notice that the outputs of Figure 6.10 and those of Figure 6.1 are spike time equivalent, as long as the RKs K_1^a and K_2^a have large enough bandwidth. In what follows we will evaluate the four Volterra terms, i.e., the four dendritic currents feeding the BSG of Neuron 1 in Figure 6.10.

Formally, the first order (feedforward) Volterra term can be written as [105]

$$\int_{\mathbb{D}_1} h_1^{s|si}(t-s)u_1(s)ds = \int_{\mathbb{D}_1} u_1(t-s)(\mathcal{P}_1^s h_1^{s|si})(s)ds. \tag{6.34}$$

Similarly, the second order (feedforward) Volterra term amounts to

$$\int_{\mathbb{D}_2} h_2^{s|si}(t-s_1,t-s_2)u_2(s_1,s_2)ds_1ds_2 = \int_{\mathbb{D}_2} u_2(t-s_1,t-s_2)(\mathcal{P}_2^s h_2^{s|si})(s_1,s_2)ds_1ds_2. \tag{6.35}$$

The above equations suggest that the projections of the feedforward kernels can be re-interpreted as inputs, whereas the signals u_1 and u_2 can be treated as feedforward kernels.

In Section 6.1.2.2 we introduced two RKHSs, $\mathcal{H}_1^a$ and $\mathcal{H}_2^a$, for modeling two different spaces of spikes. The elements of $\mathcal{H}_1^a$ are functions defined over the domain $[0, S_t^a]$ with

$$S_t^a \geq \text{supp}\{h_1^{a|ji}\} + \max\{(t_{k+1}^i - t_k^i)\}_{i=1,2,k\in\mathbb{Z}}.$$

The period S_t^a is large enough to ensure that any spike that arrives $\text{supp}\{h_1^{a|ji}\}$ seconds prior to the arrival of t_k^i, or earlier, will not affect the output of the feedback kernel on the inter-spike time interval $[t_k^i, t_{k+1}^i]$. Thus such spikes will not introduce additional error terms in the t-transform evaluated on the inter-spike time interval $[t_k^i, t_{k+1}^i]$. Note that the domain $[0, S_t^a]$ of the functions in $\mathcal{H}_1^a$ may not be the same as the domain of the input space $\mathcal{H}_1^s$. However, such a domain can be shifted on a spike by spike basis to $[t_{k+1}^i - S_t^a, t_{k+1}^i]$ for the

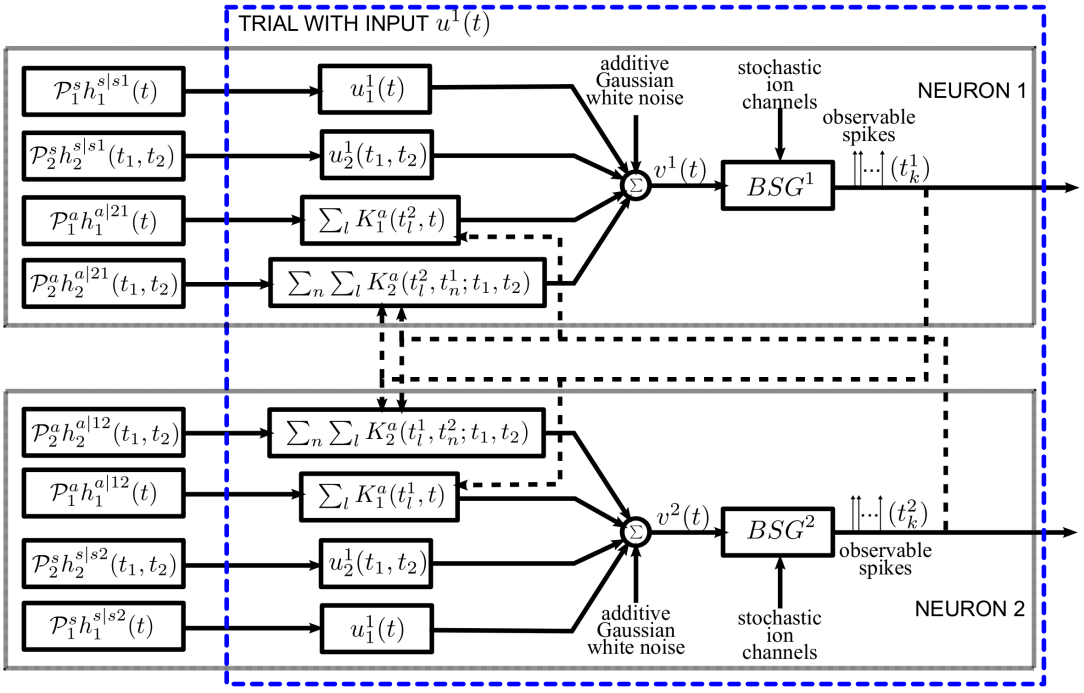


Figure 6.10: Diagram of the neural circuit that is spike timing equivalent with the one in Figure 6.1 highlighting the duality between neural decoding and functional identification. Note that the input stimuli and the DSP projections are reordered to reflect that the unknowns are the DSP projections. The input stimuli $u_1^1(t)$, $u_2^1(t_1,t_2)$, and the kernel representation of spikes (see also Section 6.1.2.2) are intrinsic to the neural circuit. The DSP projections are interpreted as inputs.

inter-spike time interval $[t_k^i, t_{k+1}^i]$. This is important for mitigating the practical limitation of modeling the stimuli as periodic functions in $\mathcal{H}_1^s$.

The response of the first-order feedback term to its spiking input on the inter-spike time interval $[t_k^i, t_{k+1}^i]$ in Figure 6.10 amounts to $(i \neq j)$

$$(\mathcal{P}_1^a h_1^{a|ji})(t) * \sum_{l:t_{k+1}^i - S_t^a \leq t_l^j < t_{k+1}^i} K_1^a(t_l^j, t) = \sum_{l:t_{k+1}^i - S_t^a \leq t_l^j < t_{k+1}^i} (\mathcal{P}_1^a h_1^{a|ji})(t - t_l^j). \tag{6.36}$$

It is clear from Section 6.1.2.2 that

$$\sum_{l:t_{k+1}^i - S_t^a \leq t_l^j < t_{k+1}^i} (\mathcal{P}_1^a h_1^{a|ji})(t - t_l^j) \xrightarrow{\mathbb{L}^2} \sum_{l:t_{k+1}^i - S_t^a \leq t_l^j < t_{k+1}^i} h_1^{a|ji}(t - t_l^j)$$

if Ω_t^a is at least larger than the effective bandwidth of $h_1^{a|ji}$ and $L_t^a \to \infty$.

Similarly, the response of the second-order feedback kernel to its spiking input on the inter-spike time interval $[t_k^i, t_{k+1}^i]$ amounts to

$$\begin{aligned} &\sum_{l:t_{k+1}^i - S_t^a \leq t_l^j < t_{k+1}^i} \sum_{n:t_{k+1}^i - S_t^a \leq t_n^i < t_{k+1}^i} (\mathcal{P}_2^a h_2^{a|ji}) \left(t - t_l^j, t - t_n^i\right) \\ \xrightarrow{\mathbb{L}^2} &\sum_{l:t_{k+1}^i - S_t^a \leq t_l^j < t_{k+1}^i} \sum_{n:t_{k+1}^i - S_t^a \leq t_n^i < t_{k+1}^i} h_2^{a|ji} \left(t - t_l^j, t - t_n^i\right) \end{aligned} \tag{6.37}$$

if Ω_t^a is large enough and $L_t^a \to \infty$.

Combining (6.34), (6.35), (6.36) and (6.37), for each spike interval $[t_k^i, t_{k+1}^i]$, the aggregated output current of the DSPs of Neuron i in Figure 6.10, shall converge to the DSP aggregated output current of Neuron i in Figure 6.1 for large enough Ω_t^a. A direct consequence of this equivalence is that, under the same additive Gaussian white noise and channel noise in the BSGs, the t-transform of the circuit in Figure 6.10 and in Figure 6.1 are identical.

Note that the outputs of the feedforward kernels are always equivalent; the equivalence of the outputs of the feedback kernels requires, however, the use of large enough bandwidth Ω_t^a. Otherwise, the equivalence in the t-transform is invalid and an additional noise term appears in the t-transform of the Neuron 1 in Figure 6.10.

The projections of the Volterra DSP kernels of Figure 6.10 are interpreted as inputs, while the input stimuli and the train of RKs at spike times replace the impulse response of the

corresponding filters. Therefore, the functional identification problem has been transformed into a dual decoding problem, where the objects to decode are the set of projections of Volterra DSP kernels and the neural circuit is comprised of "stimulus DSP kernels" and "spike DSP kernels" with the same BSGs and noise sources. The only difference is that, instead of a Single-Input Multi-Output decoding problem, the identification was transformed into a Multi-Input Multi-Output decoding problem. In addition, multiple trials using different stimuli are needed; this procedure is illustrated in block diagram form in Figure 6.11. By stimulating the neural circuit with multiple stimuli in the functional identification setting, multiple neural circuits effectively encode the projections of the DSP kernels.

We are now in the position to derive the t-transform of Neuron 1 in Figure 6.10. Assuming that $m = 1, \cdots, M$, trials are performed for identification, the t-transform (6.26) can be written as

$$ {}^m\mathcal{L}_{1k}^{s|i}[\mathcal{P}_1^s h_1^{s|si}] + {}^m\mathcal{L}_{2k}^{s|i}[\mathcal{P}_2^s h_2^{s|si}] + {}^m\mathcal{L}_{1k}^{a|i}[\mathcal{P}_1^a h_1^{a|ji}] + {}^m\mathcal{L}_{2k}^{a|i}[\mathcal{P}_2^a h_2^{a|ji}] = {}^m q_k^i + {}^m\epsilon_k^i, \qquad (6.38) $$

for $k \in {}^m\mathbb{I}^i, i, j = 1, 2, i \neq j$. Here ${}^m\mathcal{L}_{1k}^{s|i} : \mathcal{H}_1^s \to \mathbb{R}$, ${}^m\mathcal{L}_{2k}^{s|i} : \mathcal{H}_2^s \to \mathbb{R}$ are bounded linear functionals associated with the feedforward DSP kernels, and ${}^m\mathcal{L}_{1k}^{a|i} : \mathcal{H}_1^2 \to \mathbb{R}$, ${}^m\mathcal{L}_{2k}^{a|i} : \mathcal{H}_2^a \to \mathbb{R}$ are bounded linear functionals associated with the feedback DSP kernels for each trial m. The above functionals are defined as

$$ {}^m\mathcal{L}_{1k}^{s|i}[\mathcal{P}_1^s h_1^{s|si}] = \int_{{}^m t_k^i}^{{}^m t_{k+1}^i} {}^m\varphi_k^i(s) \int_{\mathbb{D}_1} u_1^m(s-r)(\mathcal{P}_1^s h_1^{s|si})(r) dr ds, $$

$$ {}^m\mathcal{L}_{2k}^{s|i}[\mathcal{P}_2^s h_2^{s|si}] = \int_{{}^m t_k^i}^{{}^m t_{k+1}^i} {}^m\varphi_k^i(s) \int_{\mathbb{D}_2} u_2^m(s-r_1, s-r_2)(\mathcal{P}_2^s h_2^{s|si})(r_1, r_2) dr_1 dr_2 ds, $$

$$ {}^m\mathcal{L}_{1k}^{a|i}[\mathcal{P}_1^a h_1^{a|ji}] = \sum_{l:{}^m t_{k+1}^i - S_t^a \leq {}^m t_l^j < {}^m t_{k+1}^i} \int_{{}^m t_k^i}^{{}^m t_{k+1}^i} {}^m\varphi_k^i(s)(\mathcal{P}_1^a h_1^{a|ji}) \left(s - {}^m t_l^j\right) ds $$

$$ {}^m\mathcal{L}_{2k}^{a|i}[\mathcal{P}_2^a h_2^{a|ji}] $$

$$ = \sum_{l:{}^m t_{k+1}^i - S_t^a \leq {}^m t_l^j < {}^m t_{k+1}^i} \sum_{n:{}^m t_{k+1}^i - S_t^a \leq {}^m t_n^i < {}^m t_{k+1}^i} \int_{{}^m t_k^i}^{{}^m t_{k+1}^i} \left[{}^m\varphi_k^i(s) \cdot (\mathcal{P}_2^a h_2^{a|ji})(s - {}^m t_l^j, s - {}^m t_n^i) \right] ds $$

$$ {}^m q_k^i = ({}^m b_k^i - {}^m I^i) \int_{{}^m t_k^i}^{{}^m t_{k+1}^i} {}^m\varphi_k^i(s) ds, $$

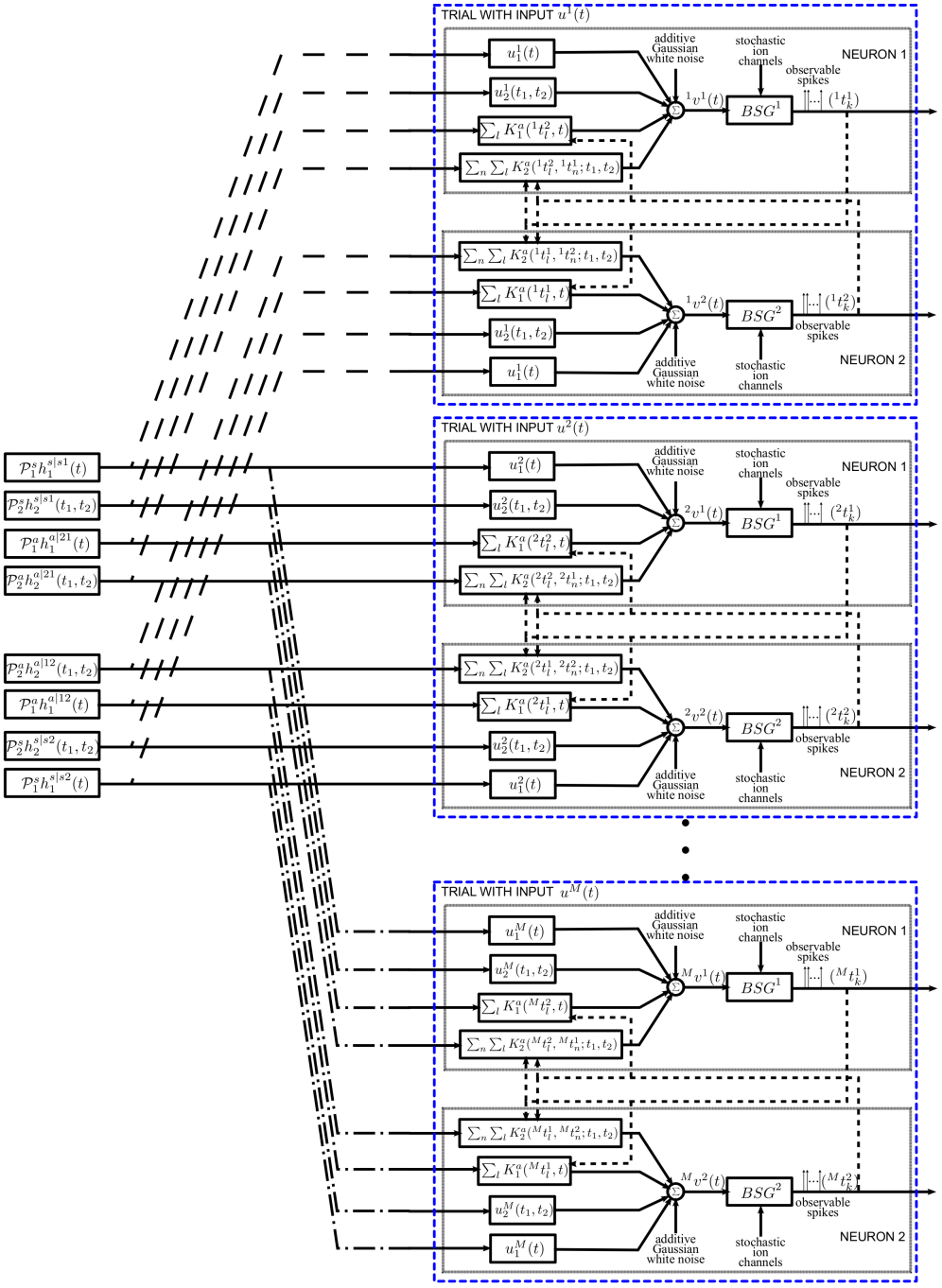


Figure 6.11: Diagram of the functional identification with multiple trials. The neural circuit is presented a different stimulus $u_1^m(t)$ for each trial m. See also Figure 6.10 for details of a single trial.

$$ {}^{m}\varphi_k^i(t) = \frac{\psi_1^i(t - {}^{m}t_k^i, {}^{m}b_k^i)}{\left(\mathbb{E}\left[{}^{m}\varepsilon_k^i\right]^2\right)^{\frac{1}{2}}} $$

and ${}^{m}\epsilon_k^i, k \in {}^{m}\mathbb{I}^i, i = 1, 2, m = 1, \cdots, M$, are independent random variables with normal distribution $\mathcal{N}(0, 1)$.

The functional identification of the neural circuit in Figure 6.10 can then be similarly defined to the decoding problem. We formulate the identification of the noisy neural circuit again as two smoothing spline problems, one for each neuron,

$$ \begin{bmatrix} \widehat{(\mathcal{P}_1^s h_1^{s|s1})} \\ \widehat{(\mathcal{P}_2^s h_2^{s|s1})} \\ \widehat{(\mathcal{P}_1^a h_1^{a|21})} \\ \widehat{(\mathcal{P}_2^a h_2^{a|21})} \end{bmatrix} $$

$$ = \underset{\left\{ \begin{smallmatrix} \mathcal{P}_1^s h_1^{s|s1} \in \mathcal{H}_1^s, \mathcal{P}_2^s h_2^{s|s1} \in \mathcal{H}_2^s \\ \mathcal{P}_1^a h_1^{a|21} \in \mathcal{H}_1^a, \mathcal{P}_2^a h_1^{a|21} \in \mathcal{H}_2^a \end{smallmatrix} \right\}}{\operatorname{argmin}} \left\{ \begin{aligned} &\sum_{r=1}^{2} \lambda_r^s \|\mathcal{P}_r^s h_r^{s|s1}\|^2 + \sum_{r=1}^{2} \lambda_r^a \|\mathcal{P}_r^a h_r^{a|21}\|^2 \\ &+ \sum_{m=1}^{M} \sum_{k \in {}^{m}\mathbb{I}^1} \left((\sum_{r=1}^{2} {}^{m}\mathcal{L}_{rk}^{s|s1}[\mathcal{P}_r^s h_r^{s|s1}] + {}^{m}\mathcal{L}_{rk}^{a|21}[\mathcal{P}_r^a h_r^{a|21}]) - {}^{m}q_k^1 \right)^2 \end{aligned} \right\} \tag{6.39} $$

and

$$ \begin{bmatrix} \widehat{(\mathcal{P}_1^s h_1^{s|s2})} \\ \widehat{(\mathcal{P}_2^s h_2^{s|s2})} \\ \widehat{(\mathcal{P}_1^a h_1^{a|12})} \\ \widehat{(\mathcal{P}_2^a h_2^{a|12})} \end{bmatrix} $$

$$ = \underset{\left\{ \begin{smallmatrix} \mathcal{P}_2^s h_1^{s|s2} \in \mathcal{H}_1^s, \mathcal{P}_2^s h_2^{s|s2} \in \mathcal{H}_2^s \\ \mathcal{P}_1^a h_1^{a|12} \in \mathcal{H}_1^a, \mathcal{P}_2^a h_1^{a|12} \in \mathcal{H}_2^a \end{smallmatrix} \right\}}{\operatorname{argmin}} \left\{ \begin{aligned} &\sum_{r=1}^{2} \lambda_r^s \|\mathcal{P}_r^s h_r^{s|s2}\|^2 + \sum_{r=1}^{2} \lambda_r^a \|\mathcal{P}_r^a h_r^{a|12}\|^2 \\ &+ \sum_{m=1}^{M} \sum_{k \in {}^{m}\mathbb{I}^1} \left((\sum_{r=1}^{2} {}^{m}\mathcal{L}_{rk}^{s|s2}[\mathcal{P}_r^s h_r^{s|s2}] + {}^{m}\mathcal{L}_{rk}^{a|12}[\mathcal{P}_r^a h_r^{a|12}]) - {}^{m}q_k^1 \right)^2 \end{aligned} \right\} \tag{6.40} $$

where ${}^{m}\mathbb{I}^i$ is the index set of spikes generated by Neuron i in trial m.

The solution can be obtained in a similar way as in Theorem 5.

Theorem 6. *The solutions to* (6.39) *is*

$$\widehat{(\mathcal{P}_1^s h_1^{s|s1})}(t) = \sum_{m=1}^{M} \sum_{k\in {}^m\mathbb{I}^1} {}^m c_k \, {}^m\phi_{1k}(t)$$

$$\widehat{(\mathcal{P}_2^s h_2^{s|s1})}(t_1,t_2) = \sum_{m=1}^{M} \sum_{k\in {}^m\mathbb{I}^1} {}^m c_k \, {}^m\phi_{2k}(t_1,t_2)$$

$$\widehat{(\mathcal{P}_1^a h_1^{a|21})}(t) = \sum_{m=1}^{M} \sum_{k\in {}^m\mathbb{I}^1} {}^m c_k \, {}^m\phi_{3k}(t)$$

$$\widehat{(\mathcal{P}_2^a h_2^{a|21})}(t_1,t_2) = \sum_{m=1}^{M} \sum_{k\in {}^m\mathbb{I}^1} {}^m c_k \, {}^m\phi_{4k}(t_1,t_2),$$

where

$$\mathbf{c} = \left[{}^1c_1 \cdots {}^1c_{|{}^1\mathbb{I}^1|}, \cdots, \cdots, {}^Mc_1 \cdots {}^Mc_{|{}^M\mathbb{I}^1|} \right]^{\mathsf{T}},$$

is the solution to the system of linear equations

$$\left((\mathbf{\Phi}_1 + \mathbf{\Phi}_2 + \mathbf{\Phi}_3 + \mathbf{\Phi}_4)^2 + \lambda_1^1 \mathbf{\Phi}_1 + \lambda_1^2 \mathbf{\Phi}_2 + \lambda_2^1 \mathbf{\Phi}_3 + \lambda_2^2 \mathbf{\Phi}_4 \right) \mathbf{c} = (\mathbf{\Phi}_1 + \mathbf{\Phi}_2 + \mathbf{\Phi}_3 + \mathbf{\Phi}_4)\, \mathbf{q},$$

where

$$\mathbf{q} = \left[{}^1q_1^1 \cdots {}^1q^1_{|{}^1\mathbb{I}^1|}, \cdots, \cdots, {}^Mq_1^1 \cdots {}^Mq^1_{|{}^M\mathbb{I}^1|} \right]^{\mathsf{T}},$$

and

$$\mathbf{\Phi}_i = \begin{bmatrix} \mathbf{\Phi}_i^{11} & \cdots & \mathbf{\Phi}_i^{1M} \\ \vdots & \ddots & \vdots \\ \mathbf{\Phi}_i^{M1} & \cdots & \mathbf{\Phi}_i^{MM}, \end{bmatrix}$$

and finally

$$[\mathbf{\Phi}_i^{mn}]_{kl} = \langle {}^m\phi_{ik}, {}^n\phi_{il} \rangle, \text{ for all } k \in {}^m\mathbb{I}^1, l \in {}^n\mathbb{I}^1, m, n = 1, 2, \cdots, M, i = 1, 2, 3, 4.$$

In addition, the sampling functions ${}^m\phi_{ik}$ *are given by*

$${}^m\phi_{1k}(t) = {}^m\mathcal{L}_{1k}^{s|1} K^s_{1|t},$$

$${}^m\phi_{2k}(t_1,t_2) = {}^m\mathcal{L}_{1k}^{s|1} K^s_{2|t_1,t_2},$$

$${}^m\phi_{3k}(t) = {}^m\mathcal{L}_{1k}^{a|1} K^a_{1|t},$$

$${}^m\phi_{4k}(t_1,t_2) = {}^m\mathcal{L}_{2k}^{a|1} K^a_{2|t_1,t_2}.$$

Proof: The proof is similar to the one of Theorem 5. □

Since each of the kernel projections may be in a different RKHS, and their domain may also be different, the identification of all filters resemble that of the multi-sensory Time Encoding Machines. Recall that multi-sensory TEMs encode within the same circuit time-varying and space-time varying sensory signals while decoding remains tractable [100]. The solution to (6.40) can similarly be obtained as the solution to (6.39) above.

Note that we are only able to identify the projection of the Volterra kernels. This is because, in practice, we can only probe the system with signals in a bandlimited space. By increasing the bandwidth of the elements of the Hilbert space, the projection of the kernels will converge to their original form [102].

Remark 12. *It is important to note that in order to have a good estimate of the kernels, stimuli must fully explore all input spaces. This can be quite easily achieved for the feedforward DSP kernels by using many (randomly generated) signals that cover the entire frequency spectrum. However, to properly identify the feedback DSP kernels, spike trains must be diverse enough to sample its different frequency components. This may not be easy to realize in practice. For first order feedback kernels, spike trains with constant spike intervals are, for example, undesirable. The Fourier transform of regular Dirac-delta pulses is a train of Dirac-delta pulses in the Fourier domain. This means that only certain frequency responses of the DSP kernels are, for example the DC component, sampled. The rest of the frequency components are essentially in the null space of the sampling functions* ${}^m\phi_{ik}, i = 1, 2, m = 1, \cdots, M$. *Similar effect applies to the space of trigonometric polynomials. If the spike intervals exhibit small variations, many of the frequency components may be sampled but the energy at DC may be too dominant. In this case, noise may contaminate more severely the measurement of non-DC components and may yield unsatisfactory identification. This effect may pose even more stringent requirements on the identification of the second order feedback kernels, as it requires the interaction between two spike trains.*

6.3.2 Effect of Noise on Identification

In order to evaluate the effect of noise on the identification of the neural circuit in Figure 6.1 we included intrinsic noise into the example neural circuit discussed under noiseless conditions in example 3. Randomly generated signals were used in the identification examples given here. Chosen in the same way as in the decoding example in Section 6.2.3.2 all these signals are used here to identify the neuron in question. Therefore, in this section, multiple signals are used in repeat experiments to identify the parameters of a neural circuit. By contrast in Section 6.2.3.2, multiple neurons are used to encode a single signal.

First, we evaluated the effect of noise on the quality of identification of DSP kernels of Neuron 1 in Figure 6.10 with a BSG modeled by the SDE (A.4) with $10\sigma_1^i = 10\sigma_2^i = \sigma_3^i = \sigma_4^i = \sigma$. Figure 6.12 shows the SNR of the identified DSP kernels in Figure 6.10 across several noise levels σ. As expected, the general trend for all four kernels is decreasing SNR with increasing noise levels. We notice that the identified feedforward DSP kernels have similar shape as the original kernel, even at high noise levels. However, the feedback DSP kernels undergo a change in shape at high noise levels. We can see that the time instance of the peak amplitude in the first order feedback kernel is shifted to an earlier time instance.

Second, we investigated the identification of DSPs for a BSG noise model already described in Section 6.2.3.3. Figure 6.13 shows the SNR of the identified DSP kernels across a different number of sodium channels N_{Na} while $N_K = 0.3N_{Na}$. The SNR plots suggest that the identification quality increases as more ion channels are present in the BSGs.

Additionally, as discussed in Remark 12, BSG noise sources may degrade severely the identification of feedback kernels when the spike trains generated in trials are not sufficient for exploring the two spike input spaces. We show an example of the later in Figure 6.14. The two BSGs have higher bias currents and lower input current magnitude. The later was achieved by scaling down the magnitude of the DSP kernels. The combined effect results in regular spiking intervals in both neurons. The identification result under *noiseless conditions* is shown in Figure 6.14. Note that since the t-transform of the Hodgkin-Huxley neuron is not exact, a small error is introduced even if intrinsic noise is not present. We see that the feedforward DSP kernels can be identified quite well, yielding SNRs of around

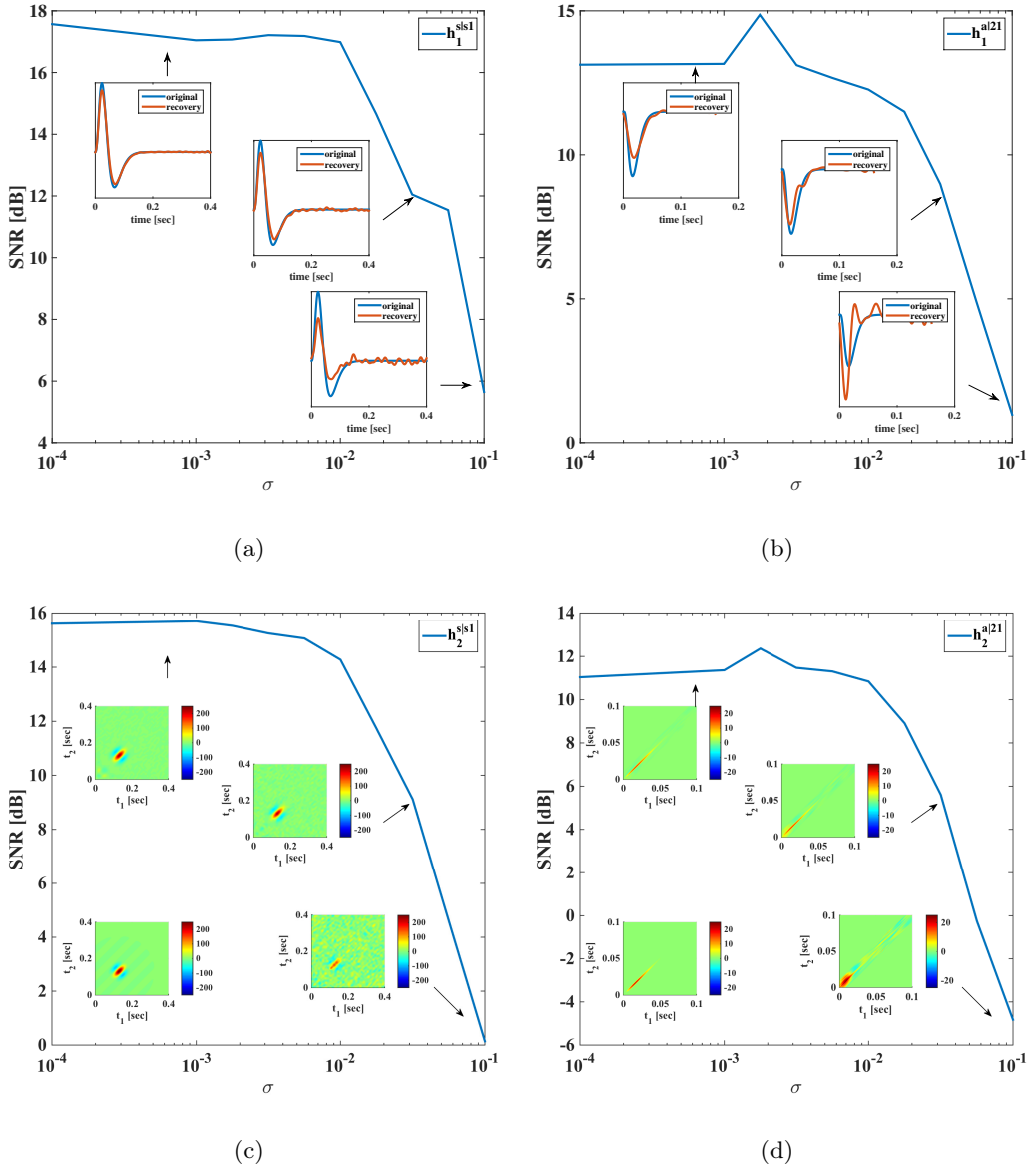


Figure 6.12: SNR of identified DSP kernels. Noise added using SDE (A.4), with $10\sigma_1^i = 10\sigma_2^i = \sigma_3^i = \sigma_4^i = \sigma$. (a) Kernel $h_1^{s|s1}$. In-sets provide a comparison between the original and the identified kernel. (b) Kernel $h_2^{s|s1}$. In-sets are identified kernels. Original kernel is on the lower left. (c) Kernel $h_1^{a|21}$. In-sets provide a comparison between the original and the identified kernel. (d) Kernel $h_2^{a|21}$. In-sets are identified kernels. Original kernel is on the lower left.

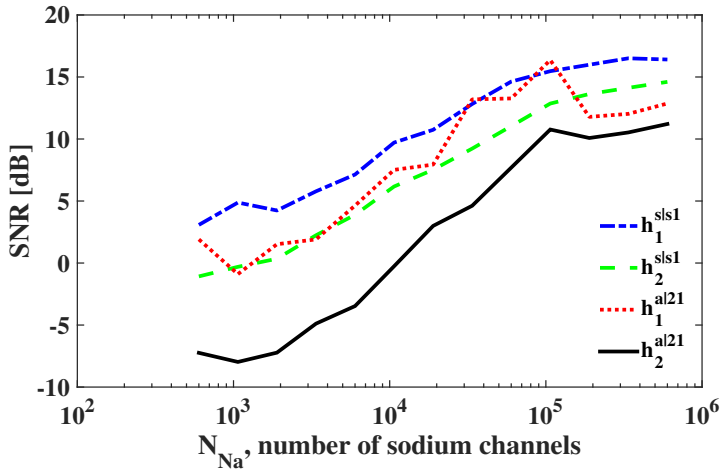


Figure 6.13: SNR of identified DSP kernels. The BSG is described by the Hodgkin-Huxley equations with a finite number of ion channels and $N_K = 0.3N_{Na}$.

17 [dB]. However, the feedback DSP kernels are not well identified. In particular, the identified second-order feedback kernel has a wide spread, similar to the high noise case in Figure 6.12(d). This suggest that the spike pattern is not sufficiently exploring the space of feedback kernels. A large number of frequency components are only weakly sampled and they can be very easily contaminated by noise. The presence of both intrinsic noise sources can exacerbate the condition further. This effect is confirmed with a simulation of the circuit using Integrate-and-Fire (IAF) neurons. Since the t-transform for the IAF neuron is exact [106], both feedback kernels can be identified even if the generated spikes only weakly explore certain frequency components. However, by artificially adding a small measurement error to the t-transform of the circuit with IAF neurons, similar results to those in Figure 6.14 can be obtained (data not shown).

We provide additional results in Appendix D. We show in Appendix D.1 the identification of DSPs of both neuron 1 and neuron2 under noiseless condition, and demonstrate that quality of identification increases as the number of spikes used in the identification increases. We also show perfect identification when the spike generators of the circuit are IAF neurons in Figure D.3. Furthermore, we show in Figure D.4 that the reconstruction quality increases when stimuli of larger bandwidth are used in identification.

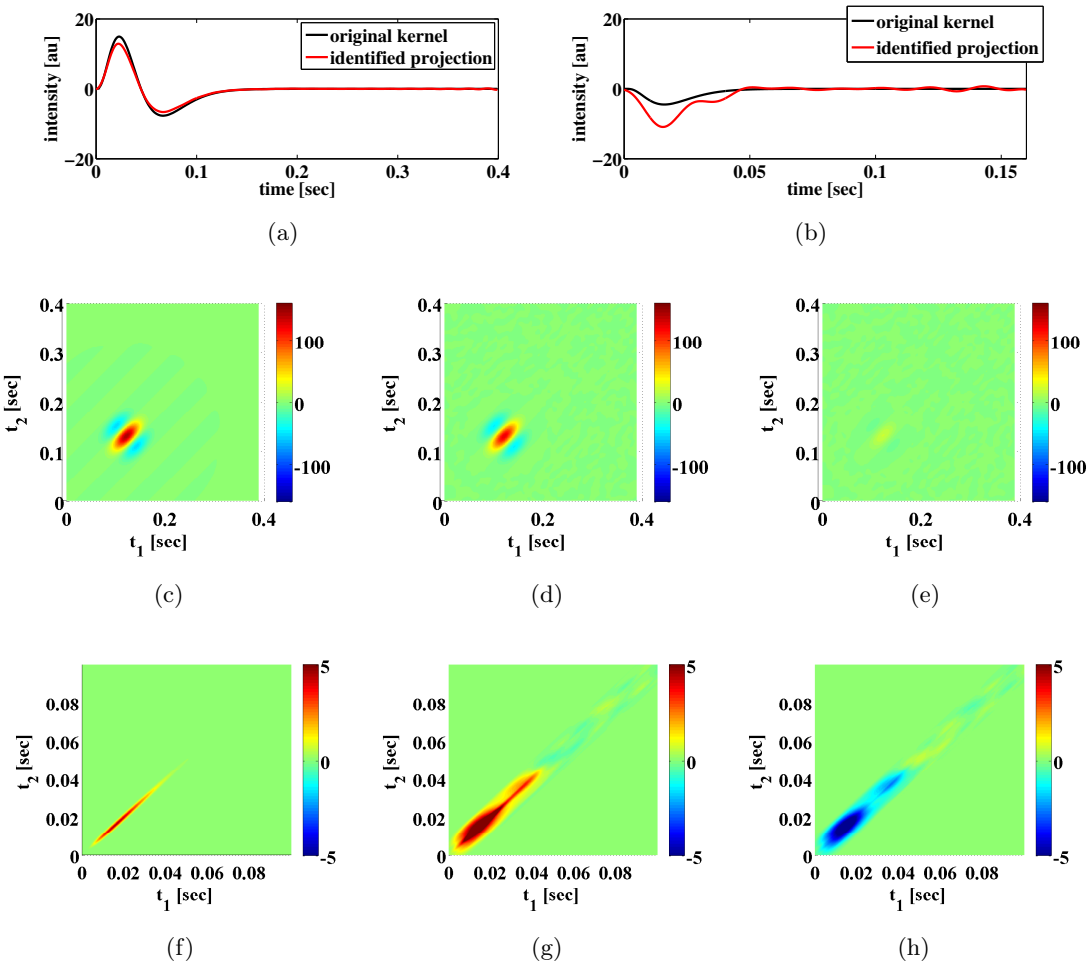


Figure 6.14: Examples of functional identification when the generated spikes do not fully explore the space of feedback kernels. (a) Original first order feedforward kernel (black) and identified projection of the kernel (red). (b) Original first order feedback kernel (black) and identified projection of the kernel (red). (c) Original second order feedforward kernel. (d) Identified projection of second order feedforward kernel. (e) Error of identified second order feedforward kernel. (f) Original second order feedback kernel. (g) Identified projection of second order feedback kernel. (h) Error of identified second order feedback kernel.

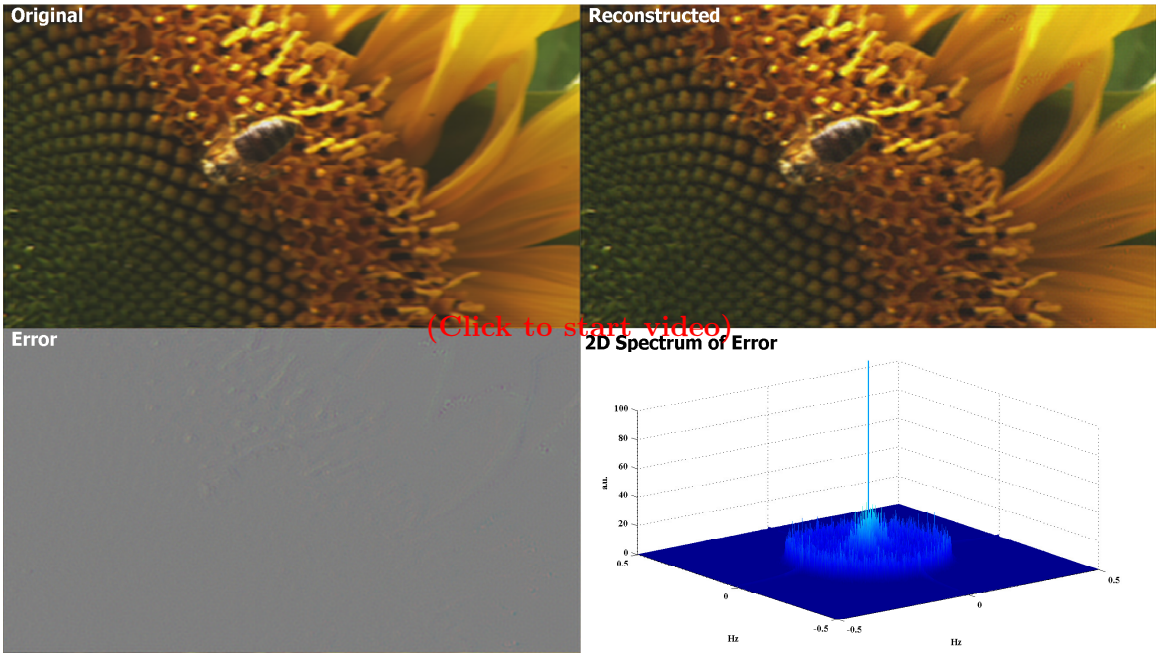
6.4 Reconstruction of Visual Stimuli under Noisy Conditions

Intrinsic noise sources in BSGs can be similarly introduced into the Video TEM we advanced in Chapter 3. Here, we show a few examples of reconstruction of natural visual stimuli encoded by the massively parallel neural circuit with intrinsic noise sources.

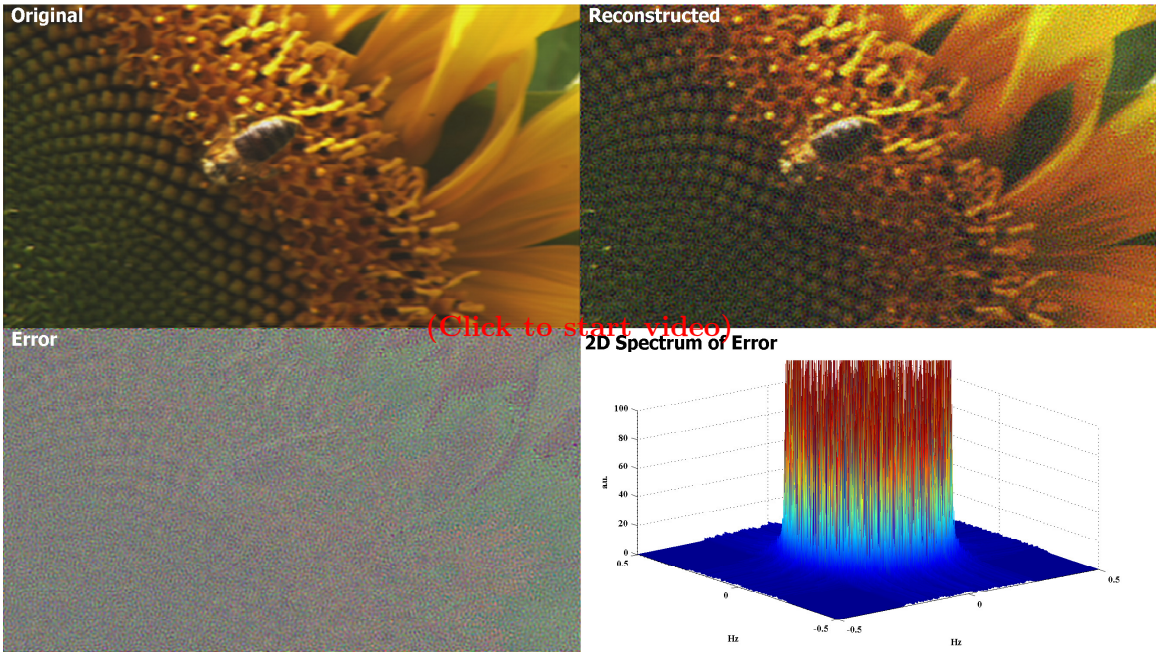
We introduce noise sources in the Hodgkin-Huxley neurons in a similar method as in the examples in Section 6.2.3. The reconstruction quality of visual stimuli decreases as the noise becomes stronger. We show this visually using examples in Section 3.5 but with two different levels of noise. The first level of noise is $\sigma_2 = 0.01, \sigma_3 = \sigma_4 = 0.1$. The reconstruction is shown in the video in Figure 6.15(a). The SNR for recovered R, G and B channels are 33.8 [dB], 31.7 [dB] and 25.8 [dB], respectively. Mean SSIM index values are 0.983, 0.981, 0.974, respectively. We further increased the noise level by 10 fold, yielding a visibly noisy reconstruction. The reconstruction is shown in Figure 6.15(b). The SNR for recovered R, G and B channels are 17.8 [dB], 15.2 [dB] and 5.94 [dB], respectively. Mean SSIM index values are 0.526, 0.466, 0.295, respectively. These examples demonstrate that the decoding algorithm we developed in Section 3.3 can process a multidimensional spike train generated by a massively parallel encoding system with intrinsic noise sources.

6.5 Discussion and Summary

In this chapter, we introduced a novel neural circuit architecture based on a neuron model with a biophysical mechanism of spike generation and feedforward as well as feedback dendritic stimulus processors with intrinsic noise sources. Under this architectural framework, we quantitatively studied the effect of intrinsic noise on dendritic stimulus processing and on spike generation. We investigated how intrinsic noise sources affect the stimulus representation by decoding encoded stimuli from spikes, and quantified the effect of noise on the functional identification of neural circuits. We argued that a duality between stimulus decoding and functional identification holds. Therefore, the encoding framework based on the neural circuit architecture studied here can be applied to both the reconstruction of the encoded signal and the identification of the dendritic stimulus processors. We systematically showed how the precision in decoding is affected by different levels of stochastic variability



(a)



(b)

Figure 6.15: Examples of a reconstructed visual scene encoded by neural circuits with intrinsic noise sources. (a) Encoding circuit with weak noise: $\sigma_2 = 0.01, \sigma_3 = \sigma_4 = 0.1$. (b) Encoding circuit with stronger noise: $\sigma_2 = 0.1, \sigma_3 = \sigma_4 = 1$. (See also Figure 3.13 for a description of each panel) (From [110] with permission, ©2014 IEEE)

within the circuit. These results apply verbatim to the functional identification of dendritic stimulus processors via the key duality property mentioned above.

Our theoretical framework highlights two indispensable components of modeling signal representation/ processing in a neural circuit - dendritic stimulus processing and spike generation. Such a divide and conquer strategy is ubiquitous in engineering circuits and leads to a separation of concerns. Recent experimental studies also showed that interesting nonlinear processing effects take place first in the dendritic trees rather than in the axon hillock [202].

We presented here two types of non-linear dendritic stimulus processors. The first type are feedforward DSPs that respond to continuous input sensory stimuli. The second type are feedback DSPs that respond to *spiking* inputs. We quantitatively demonstrated how intrinsic noise sources would affect the identification quality of all these DSPs. The examples in Section 6.3.2 suggest that in identification feedback kernels are more vulnerable to internal noise sources than feedforward kernels. In particular, the overall shape of the identified feedback kernels differs significantly from that of the underlying kernels when the strength of noise sources becomes large. Meanwhile the identified feedforward kernels are qualitatively preserved, albeit not accurately.

Most of the single neuron models (LIF, QIF) in the literature have focused on the spike generation mechanism. The encoding capability of these models is typically investigated based on rate encoding [44, 127, 144]. For both decoding and identification we used here the occurrence times of spikes generated by spiking neuron models. Most importantly, the BSG models discussed here were characterized by a PRC manifold [88] in the presence of noise, while many simplified models (such as the LIF) can be effectively described with a single PRC. Other sensory neuron models, *e.g.,* GLM [150], usually rely on a rate-based output or Poisson spike generation that do not take into account key advances in dynamical systems-based spiking neuron models.

As already mentioned before, we investigated how intrinsic noise sources affect the stimulus representation by decoding encoded stimuli from spikes. We are not suggesting, however, that the decoding algorithm considered here is implemented in the brain. Rather, we argue that decoding is effective in measuring how well information is preserved in the spike domain. The decoding formalism allowed us to investigate how noise affects the fidelity of

signal representation by a population of neurons by reconstructing stimuli and comparing their information content in the stimulus space.

While decoding can serve as an "oscilloscope" in understanding stimulus representation in sensory systems, functional identification serves as a guide in experiments to functionally identify sensory processing. Based on spike times, the identification algorithm presents a clear bound on the number of spikes that are necessary for perfect identification under noiseless conditions. Phrased differently, when a certain number of spikes are acquired from a neuron of interest, the identification algorithm places a constraint on the maximum DSP kernel bandwidth that can perfectly be recovered.

In more practical terms, we advanced two important applications of the circuit architecture considered in this chapter. The first one considers dendritic stimulus processors that process information akin to complex cells in V1. The second one adapts the widely used Hodgkin-Huxley model known in the context of neural excitability [81] and analysis of neuronal stochastic variability to stimulus encoding in the presence of noise.

Based on the rigorous formalism of TEMs [106], we extended our previous theoretical framework [112] and argued that spike timing is merely a form of generalized sampling of stimuli. By studying sampling (or measurements) in the presence of intrinsic noise sources, we showed to what extent neurons can represent sensory stimuli in noisy environments as well as how much noise the identification process can tolerate while preserving an accurate understanding of circuit dynamics.

The reconstruction and identification quality are certainly not only related to the strength of noise, but also the strength of the signal. In particular, when the signal strength is small, two facts may affect the quality of reconstruction. First, neurons may not produce enough spikes that have valid t-transforms. Second, they may be contaminated by even weak noise, *i.e.*, the signal-to-noise ratio is low. It is well known, however, that neural systems use gain control to boost the relevant signal [59, 171, 191]. Such strategy may be useful for increasing the signal strength relatively to the strength of the noise. Gain control may also suppress large signals to fit into the range of operation of the BSGs. The gain control itself, maybe considered as a type of Volterra feedforward DSP kernel [105] and the interaction with feedback loops driven by spikes. The lack of spikes may be compensated by adding

other neurons that are sensitive to other features in the input stimuli.

A key feature in our neural circuit model is the nonlinear processing in the feedforward and feedback paths. Nonlinear interaction between feedforward DSPs and feedback DSPs have not been considered here. However, they are of interest and could be addressed in the future. Self-feedback was not included in the model for clarity, but can be considered within the framework of our approach. Self-feedback was introduced to add refractoriness to phenomenological neuron models [87, 149]. Our BSG models, on the contrary, are conductance-based models that have a refractory period built in.

Throughout this chapter we assumed that the BSGs themselves have been perfectly identified. The intrinsic noise in the BSGs may degrade the identification quality of conditional PRCs. This may result in a lower identification quality as shown in the examples. It is beneficial to investigate in the future a method that can identify the entire circuit at once so that intrinsic noise in the circuit only affects the identification process a single time.

The theoretical results obtained here suggest a number of experiments in the early olfactory system of fruit flies. The glomeruli of the antennal lobe can be modeled using the Volterra DSPs discussed here and the projection neurons in the antennal lobe are accessible by patch clamping [107]. Functional identification of DSPs can then be carried out for studying olfactory stimulus processing in an accessible circuit with intrinsic noise sources [132].

Chapter 7

Reconstruction of Visual Stimuli Encoded by Complex Cells and Functional Identification of Energy Processing

Similar to simple cells, complex cells in the primary visual cortex (V1) are selective to oriented edges/lines over a limited region of the visual field [79]. While simple cells respond maximally to a particular phase of the edge, complex cells are largely phase invariant [162]. Therefore, the receptive fields of complex cells cannot be simply subdivided into excitatory and inhibitory regions [79]. Anatomically, complex cells do not receive input directly from the Lateral Geniculate Nucleus (LGN), but are thought to receive inputs from simple cells. Receptive fields of simple cells can be modeled as linear Gabor filters while processing in complex cells can be modeled with a quadrature pair of Gabor filters followed by squaring [1].

It is not clear, however, what is the representation capability of complex cells. Can the visual scene still be reconstructed based on the spiking output generated by an ensemble of complex cells? Can the processing taking place in complex cells be identified with high accuracy? Since the receptive fields of complex cells are non-linear, spike-triggered average

cannot be applied to identify them [22]. A few higher order statistics based algorithms have been recently proposed, including the spike-triggered covariance (STC) [147] and the minimum and maximum information methods [56]. Limitations of current theoretical models call for methods of non-linear systems identification.

As shown in Chapter 6, processing in complex cells has the complexity of Volterra models. General Volterra based models, however, run into the curse of dimensionality. We demonstrate, however, that complex cells have a processing structure that can be exploited and decoding and identification problems can be efficiently solved. We provide (i) an algorithm that reconstructs the visual stimuli based on the spike times generated by neural circuits with widely employed complex cells models [22], and (ii) propose an mechanistic algorithm for functionally identifying the processing in complex cells using the spike times they generate [119]. These algorithms are, again, based on the key observation that functional identification of processing in complex cells can be viewed as the dual problem of decoding stimuli that are encoded by an ensemble of complex cells.

7.1 Encoding of Stimuli with a Population of Complex Cells

7.1.1 A Complex Cell Time Encoding Machine

We consider a neural circuit consisting of a population of M complex cells that encode space-time stimuli (see Figure 7.1). The stimuli considered here are of the form $u_1(x,t), (x,t) \in \mathbb{D}$, where (for simplicity in presentation) the spatial domain is one-dimensional. The input stimulus $u_1(x,t)$ to neuron i is first processed by a quadrature pair of Gabor filters, squared and then linearly summed. This is known as the energy processing in the complex cell [22]. This energy processing can also be equivalently described as a Dendritic Stimulus Processor (DSP). The output of the DSP, denoted by $v^i(t)$, is then fed into the Spike Generator (SG) of neuron i. The SG encodes the DSP output into the spike train $(t_k^i)_{k\in\mathbb{Z}}$.

By denoting for each neuron $i = 1, 2, \cdots, M$, the phase-shifted (quadrature pair) Gabor

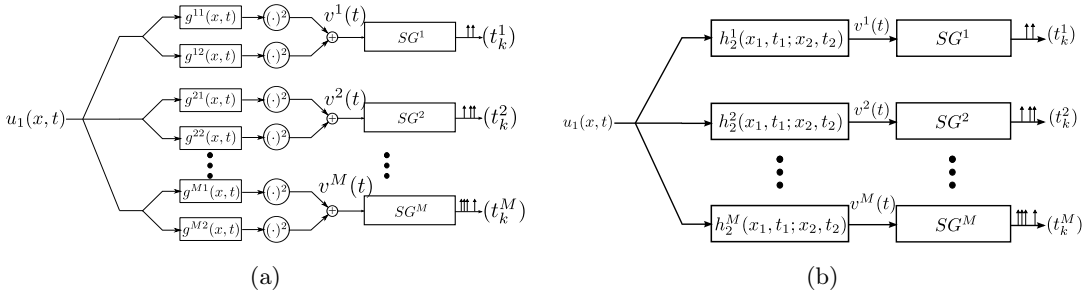


Figure 7.1: Complex Cell TEM (a) Population of complex cells, (b) Equivalent I/O circuit.

filters by $g^{i1}(x,t)$ and $g^{i2}(x,t)$, the output of the DSP amounts to

$$\begin{aligned} v^i(t) =&(\int_{\mathbb{D}} g^{i1}(x,t-s)u_1(x,s)dxds)^2 + (\int_{\mathbb{D}} g^{i2}(x,t-s)u_1(x,s)dxds)^2 \\ =&\int_{\mathbb{D}^2} h_2^i(x_1,t-s_1;x_2,t-s_2)u_1(x_1,s_1)u_1(x_2,s_2)dx_1ds_1dx_2ds_2. \end{aligned} \tag{7.1}$$

Here

$$h_2^i(x_1,t_2;x_2,t_2) = g^{i1}(x_1,t_1)g^{i1}(x_2,t_2) + g^{i2}(x_1,t_1)g^{i2}(x_2,t_2) \tag{7.2}$$

denotes the second order Volterra kernel that arises in the Volterra expansion of a nonlinear functional, $\mathbb{D}$ denotes the domain of u_1 and $\mathbb{D}^2$ the Cartesian product of the domain. The equivalent I/O circuit model is illustrated in Figure 7.1(b).

We assume the second order Volterra DSP kernel $h_2(x_1,t_1;x_2,t_2)$ to be real, bounded-input bounded-output (BIBO)-stable, causal and of finite memory. For simplicity, we will use the Integrate-and-Fire neuron model as the spike generation mechanism (see, e.g., [98]). The integration constant, bias and threshold of the IAF neuron $i = 1, 2, \cdots, M$, are denoted by C^i, b^i and δ^i, respectively. In what follows, we formulate the encoding of the stimuli by the Complex Cell TEMs.

7.1.2 Modeling of Stimuli

We model the space-time varying stimuli $u_1 = u_1(x,t)$, $(x,t) \in \mathbb{D}$, to be real-valued elements of the space of trigonometric polynomials [112].

Definition 14. *The space of trigonometric polynomials $\mathcal{H}_2^s$ is the Hilbert space of complex-*

valued functions

$$u_1(x,t) = \sum_{l_x=-L_x}^{L_x} \sum_{l_t=-L_t}^{L_t} c_{l_x l_t} e_{l_x l_t}(x,t), \tag{7.3}$$

over the domain $\mathbb{D} = [0, S_x] \times [0, S_t]$*, where*

$$e_{l_x l_t}(x,t) = e_{l_x}(x) e_{l_t}(t) = \frac{1}{\sqrt{S_x}} \exp\left(\frac{j l_x \Omega_x}{L_x} x\right) \frac{1}{\sqrt{S_t}} \exp\left(\frac{j l_t \Omega_t}{L_t} t\right).$$

Here Ω_x*,* Ω_t *denote the bandwidth, and* L_x, L_t *the order of the space in each variable. Stimuli* $u_1 \in \mathcal{H}_2^s$ *are periodic with period* $S_x = \frac{2\pi L_x}{\Omega_x}, S_t = \frac{2\pi L_t}{\Omega_t}$*.*

$\mathcal{H}_2^s$ is a Reproducing Kernel Hilbert Space (RKHS) [10] with reproducing kernel (RK)

$$K_1(x', t'; x, t) = \sum_{l_x=-L_x}^{L_x} \sum_{l_t=-L_t}^{L_t} e_{l_x l_t}(x - x', t - t'). \tag{7.4}$$

We denote the total dimension of $\mathcal{H}_2^s$ by $dim(\mathcal{H}_2^s) = (2L_x + 1)(2L_t + 1)$, the spatial dimension of $\mathcal{H}_2^s$ as $dim_x(\mathcal{H}_2^s) = 2L_x + 1$ and the temporal dimension by $dim_t(\mathcal{H}_2^s) = 2L_t + 1$. Clearly, $dim(\mathcal{H}_2^s) = dim_x(\mathcal{H}_2^s) \cdot dim_t(\mathcal{H}_2^s)$.

Definition 15. *The tensor product space* $\mathcal{H}_4^s = \mathcal{H}_2^s \otimes \mathcal{H}_2^s$ *is a Hilbert space of complex-valued functions*

$$u_2(x_1, t_1; x_2, t_2) = \sum_{l_{x_1}=-L_x}^{L_x} \sum_{l_{x_2}=-L_x}^{L_x} \sum_{l_{t_1}=-L_t}^{L_t} \sum_{l_{t_2}=-L_t}^{L_t} d_{l_{x_1} l_{t_1} l_{x_2} l_{t_2}} e_{l_{x_1} l_{t_1}}(x_1, t_1) \cdot e_{l_{x_2} l_{t_2}}(x_2, t_2) \tag{7.5}$$

over the domain $\mathbb{D}^2$*.*

$\mathcal{H}_4^s$ is an RKHS with reproducing kernel

$$\begin{aligned} &K_2(x_1', t_1', x_2', t_2'; x_1, t_1, x_2, t_2) \\ &= \sum_{l_{x_1}=-L_x}^{L_x} \sum_{l_{x_2}=-L_x}^{L_x} \sum_{l_{t_1}=-L_t}^{L_t} \sum_{l_{t_2}=-L_t}^{L_t} e_{l_{x_1} l_{t_1}}(x_1 - x_1', t_1 - t_1') \cdot e_{l_{x_2} l_{t_2}}(x_2 - x_2', t_2 - t_2'). \end{aligned} \tag{7.6}$$

Note that $dim(\mathcal{H}_4^s) = dim(\mathcal{H}_2^s)^2$.

The two RKHS's defined here are similar to those defined in Chapter 6.1.2. Note that elements in $\mathcal{H}_2^s$ are the actual inputs to the Complex Cell TEMs. We shall see in what follows that the operation of the second order Volterra kernels is linear on elements in $\mathcal{H}_4^s$.

7.1.3 Encoding and the t-Transform

We now formulate the encoding of the Complex Cell TEM.

Lemma 4. *The encoding by the Complex Cell TEM of the space-time stimulus $u_1 \in \mathcal{H}_2^s$ into the spike train sequence $(t_k^i), k \in \mathbb{I}^i, i = 1, 2, \cdots, M$ is given in functional form by*

$$\mathcal{T}_k^i u_2 = q_k^i, k \in \mathbb{I}^i, i = 1, \cdots, M, \tag{7.7}$$

where M is the total number of neurons, $\mathbb{I}^i$ is, by abuse of notation, the spike train index set of complex cell i and $\mathcal{T}_k^i : \mathcal{H}_4^s \to \mathbb{R}$, are bounded linear functionals defined by

$$\mathcal{T}_k^i u_2 = \int_{t_k^i}^{t_{k+1}^i} \int_{\mathbb{D}^2} h_2^i(x_1, t - s_1; x_2, t - s_2) u_2(x_1, s_1; x_2, s_2) dx_1 ds_1 dx_2 ds_2 dt, \tag{7.8}$$

with $u_2(x_1, t_1; x_2, t_2) = u_1(x_1, t_1) u_1(x_2, t_2)$. Finally, $q_k^i = C^i \delta^i - b^i(t_{k+1}^i - t_k^i)$.

Proof: The t-transform of the i-th IAF neuron is given by [106]

$$\int_{t_k^i}^{t_{k+1}^i} v^i(t) dt = C^i \delta^i - b^i(t_{k+1}^i - t_k^i). \tag{7.9}$$

The relationship (7.7) follows after replacing $v^i(t)$ given in (7.1) in the equation (7.9) above. □

Remark 13. *$u_2 = u_1(x_1, t_1) \cdot u_1(x_2, t_2)$ can be interpreted as a nonlinear map of the stimulus u_1 into u_2 defined in a higher dimensional space. The operation performed by the second order Volterra kernel on u_2 in (7.8) is linear. Thus, (7.7) shows that the encoding of space-time stimuli by an ensemble of complex cells can be viewed as generalized sampling.*

(7.7) is called the *t-transform* of the Complex Cell TEMs. It can be compactly written in matrix form as shown below.

Theorem 7. *The coefficients of the space-time stimulus $u_1 \in \mathcal{H}_2^s$ encoded by the Complex Cell TEMs satisfy the set of equations*

$$\mathbf{Tr}(\mathbf{\Phi}_k^i \mathbf{C}) = q_k^i, k \in \mathbb{I}^i, i = 1, \cdots, M, \tag{7.10}$$

where $\mathbf{C}$ *is a rank-*1 *Hermitian matrix with* $\mathbf{C} = \mathbf{c}\mathbf{c}^{\mathsf{H}}$*, and*

$$\mathbf{c} = [c_{-L_x,-L_t}, \cdots, c_{-L_x,L_t}, c_{-L_x+1,-L_t}, \cdots, c_{-L_x+1,L_t}, \cdots, \cdots, c_{L_x,-L_t}, \cdots, c_{L_x,L_t}]^{\mathsf{T}},$$

and $(\mathbf{\Phi}_k^i)$*,* $k \in \mathbb{I}^i, i = 1, \cdots, M$*, are Hermitian matrices and the entry at the* $(l_{x_2} + L_x + 1 + (l_{t_2} + L_t)(2L_t + 1))$*-th row,* $(l_{x_1} + L_x + 1 + (l_{t_1} + L_t)(2L_t + 1))$*-th column amounts to*

$$\begin{aligned}
&[\mathbf{\Phi}_k^i]_{l_{x_2}l_{t_2};l_{x_1}l_{t_1}} \\
&= \int_{t_k^i}^{t_{k+1}^i} e_{l_{t_1}-l_{t_2}}(t)dt \int_{\mathbb{D}^2} h_2^i(x_1, s_1; x_2, s_2) e_{l_{x_1},-l_{t_1}}(x_1, s_1) e_{-l_{x_2},l_{t_2}}(x_2, s_2) dx_1 ds_1 dx_2 ds_2,
\end{aligned} \tag{7.11}$$

Proof: Plugging in the general form of u_2 in (7.5) into (7.8), the left hand side of (7.7) amounts to

$$\begin{aligned}
&\sum_{l_{x_1}=-L_x}^{L_x} \sum_{l_{t_1}=-L_t}^{L_t} \sum_{l_{x_2}=-L_x}^{L_x} \sum_{l_{t_2}=-L_t}^{L_t} d_{l_{x_1}l_{t_1},-l_{x_2},-l_{t_2}} \\
&\int_{t_k^i}^{t_{k+1}^i} e_{l_{t_1}-l_{t_2}}(t)dt \int_{\mathbb{D}^2} h_2^i(x_1, s_1; x_2, s_2) e_{l_{x_1},-l_{t_1}}(x_1, s_1) e_{-l_{x_2},l_{t_2}}(x_2, s_2) dx_1 dx_2 ds_1 ds_2.
\end{aligned}$$

It is easy to verify that this can be written as $\mathbf{Tr}(\mathbf{\Phi}_k^i \mathbf{D})$, where the $(l_{x_1} + L_x + 1 + (l_{t_1} + L_t)(2L_t + 1))$-th row, $(l_{x_2} + L_x + 1 + (l_{t_2} + L_t)(2L_t + 1))$-th column entry of $\mathbf{D}$ amounts to $[\mathbf{D}]_{l_{x_1}l_{t_1}l_{x_2}l_{t_2}} = d_{l_{x_1}l_{t_1},-l_{x_2},-l_{t_2}}$.

The form of $\mathbf{C}$ is given by the structure of $u_2(x_1, t_1; x_2, t_2) = u_1(x_1, t_1)u_1(x_2, t_2)$, where $u_1(x, t)$ is of the form (7.3). (7.10) can be obtained by replacing $\mathbf{D}$ by $\mathbf{C}$.

We note that since $h_2^i, i = 1, \cdots, M$, are assumed to be real valued, $(\mathbf{\Phi}_k^i), k \in \mathbb{I}^i, i = 1, \cdots, M$, are Hermitian. □

Remark 14. *We point out that Theorem 7 is applicable to any second order Volterra kernel that is not limited to be of the form* (7.2)*.*

7.2 Reconstruction of Stimuli

A reconstruction of the signal u_2 is in principle possible by solving a linear set of equations. This is shown in Chapter 7.2.1. However, the cost of the algorithm is prohibitive. We show that an efficient algorithm can be constructed that exploits the structure of the neural circuit.

7.2.1 Volterra Time Decoding Machines

Assuming that the spike times $(t_k^i), k \in \mathbb{Z}, i = 1, 2, \cdots, M$, are known, by Lemma 4, the Complex Cell TEM encodes the stimulus via a set of linear functionals on u_2. Thus, the reconstruction of u_2 can *in principle* be obtained by inverting the set of linear equations (7.7). We device an algorithm that is similar to Theorem 5 but with only the second order Volterra terms and by formulating the problem directly for coefficients $d_{l_{x_1} l_{t_1} l_{x_2} l_{t_1}}$ in the trigonometric polynomials (7.5).

Algorithm 1. *The coefficients of $u_2 \in \mathcal{H}_4^s$ in (7.5) are the solution to the system of linear equations*

$$\mathbf{\Xi d} = \mathbf{q}, \quad \textit{where} \quad \mathbf{\Xi} = [(\mathbf{\Xi}^1)^\mathsf{T}, \cdots, (\mathbf{\Xi}^M)^\mathsf{T}]^\mathsf{T} \quad \textit{and} \quad \mathbf{q} = [(\mathbf{q}^1)^\mathsf{T}, \cdots, (\mathbf{q}^M)^\mathsf{T}]^\mathsf{T} \tag{7.12}$$

with $\left[\mathbf{q}^i\right]_k = q_k^i, [\mathbf{d}]_{l_{x_1} l_{t_1} l_{x_2} l_{t_2}} = d_{l_{x_1} l_{t_1} l_{x_2} l_{t_2}}$ *and*

$$\begin{aligned}
&\left[\mathbf{\Xi}^i\right]_{k; l_{x_1} l_{t_1} l_{x_2} l_{t_2}} \\
&= \int_{t_k^i}^{t_{k+1}^i} e_{l_{t_1} + l_{t_2}}(t) dt \int_{\mathbb{D}^2} h_2^i(x_1, s_1; x_2, s_2) e_{l_{x_1}, -l_{t_1}}(x_1, s_1) e_{l_{x_2}, -l_{t_2}}(x_2, s_2) dx_1 dx_2 ds_1 ds_2,
\end{aligned}$$

A necessary condition to perfect recovery is that the total number of spikes exceeds $dim(\mathcal{H}_4^s) + M = dim(\mathcal{H}_2^s)^2 + M$ and the number of neurons satisfies $M \geq \frac{dim(\mathcal{H}_4^s)}{4L_t+1} = \frac{dim(\mathcal{H}_2^s)^2}{4L_t+1}$.

Remark 15. *For 2-dimensional space-time varying stimuli, the dimension of $\mathcal{H}_2^s$ is typically large. Due to the square in the recovery condition, the number of spikes needed to faithfuly recover u_2 becomes prohibitively large .*

7.2.2 Complex Cell Time Decoding Machines

A large number of measurements are required to reconstruct $\mathbf{D}$ as described in Chapter 7.2.1. However, since $u_2(x_1, t_1; x_2, t_2) = u_1(x_1, t_1)u_1(x_2, t_2)$, $\mathbf{C}$ in (7.10) is a rank 1 matrix rather than possibly full rank as $\mathbf{D}$ in the proof of Theorem 7. Therefore, it is possible to reconstruct u_2 using fewer measurements. Here, we provide a reconstruction algorithm based on the rank minimization problem

$$\begin{aligned} \text{minimize} \quad & \mathbf{Rank}(\mathbf{C}) \\ \text{s.t.} \quad & \mathbf{Tr}(\mathbf{\Phi}_k^i \mathbf{C}) = q_k^i, k \in \mathbb{I}^i, i = 1, \cdots, M \\ & \mathbf{C} \succcurlyeq 0 \end{aligned} \tag{7.13}$$

Rank minimization has recently been applied to phase retrieval problems [20] that are closely related to decoding of stimuli encoded with Complex Cell TEMs. Complex cell DSPs process the energy of stimuli in certain frequency ranges and thereby the phase information is lost. Machines performing reconstruction of space-time varying stimuli encoded by Complex Cell TEMs based on rank minimization are called Complex Cell Time Decoding Machines (Complex Cell TDMs).

The rank minimization problem in (7.13) is NP-hard. A well known heuristic is to replace the objective in (7.13) with the trace of $\mathbf{C}$ [53].

Algorithm 2. *To reconstruct u_1 from the spike times generated by the Complex Cell TEM (i) solve the semidefinite program (SDP)*

$$\begin{aligned} \textit{minimize} \quad & \mathbf{Tr}(\mathbf{C}) \\ \textit{s.t.} \quad & \mathbf{Tr}(\mathbf{\Phi}_k^i \mathbf{C}) = q_k^i, k \in \mathbb{I}^i, i = 1, \cdots, M \\ & \mathbf{C} \succcurlyeq 0 \end{aligned} \tag{7.14}$$

and denote the optimal solution $\mathbf{C}_*$. $\mathbf{C}_*$ approximates the solution to (7.13).
(ii) perform the eigen-decomposition of $\mathbf{C}_*$. Denote the largest eigenvalue by λ_* and the corresponding eigenvector by $\mathbf{v}_*$. If (7.14) *does not exactly return a rank-*1 *matrix, choose the largest eigenvalue and disregard the rest. $\mathbf{c}_*$ can be recovered, up to a complex-valued scaling factor, by $\mathbf{c}_* = \sqrt{\lambda_*}\mathbf{v}_*$.*

(iii) the reconstructed stimulus $\hat{u}_1$ *is given by (up to a sign)*

$$\hat{u}_1(x,t) = \sum_{l_x=-L_x}^{L_x} \sum_{l_t=-L_t}^{L_t} c_{l_x l_t} e_{l_x l_t}(x,t),$$

where $\mathbf{c} = \mathbf{c}_* \cdot \frac{|[\mathbf{c}_*]_{00}|}{[\mathbf{c}_*]_{00}}$ *if* $[\mathbf{c}_*]_{00} \neq 0$ *and* $\mathbf{c} = \mathbf{c}_*$ *otherwise, and*

$$\mathbf{c} = \left[c_{-L_x,-L_t}, \cdots, c_{-L_x,L_t}, c_{-L_x+1,-L_t}, \cdots, c_{-L_x+1,L_t}, \cdots, c_{L_x,-L_t}, \cdots, c_{L_x,L_t}\right]^{\mathsf{T}}.$$

Remark 16. *Note that we can reconstruct* $u_1(x,t)$ *up to a sign, since* $\mathbf{C} = \mathbf{c}\mathbf{c}^{\mathsf{H}}$ *can be either positive semi-definite or negative semi-definite. The formalism in* (7.13) *assumes that it is the former.*

Remark 17. *Note that* (7.13) *can be alternatively solved by replacing the objective with the log-det heuristic [53], that is*

$$\begin{aligned} &\textit{minimize} && \textit{log det}(\mathbf{C} + \lambda \mathbf{I}) \\ &\textit{s.t.} && \mathbf{Tr}(\mathbf{\Phi}_k^i \mathbf{C}) = q_k^i, k \in \mathbb{I}^i, i = 1, \cdots, M \\ & && \mathbf{C} \succcurlyeq 0 \end{aligned} \tag{7.15}$$

where $\lambda > 0$ *is a small regularization constant. This optimization may further reduce the rank of* $\mathbf{C}_*$ *compared to* (7.14) *[53].*

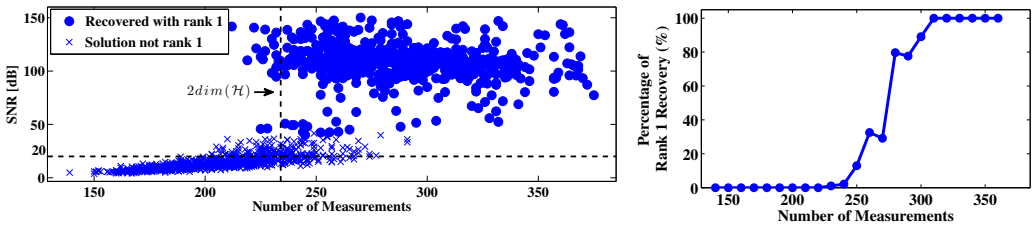
7.2.3 Example

For the simulation results presented here we used a population of 62 direction selective complex cells, *i.e.*, with spatio-temporal non-separable kernels. The DSP of each complex cell consisted of a quadrature pair of spatio-temporal Gabor filters, with dilations and translations on a dyadic grid.

The Complex Cell TEM encoded randomly generated signals in $\mathcal{H}_2^s$, with domain $\mathbb{D} = [0, 32] \times [0, 0.4]$ ([a.u.] and [sec], respectively) and $L_x = 6, L_t = 4, \Omega_x = 0.1875 \cdot 2\pi, \Omega_t = 10 \cdot 2\pi$. Thus $dim(\mathcal{H}_2^s) = 117$. We adjusted the thresholds of the IAF neurons in order to evaluate how many measurements are required for the Complex Cell TDM to perfectly

reconstruct the input stimuli. The simulations were performed in MATLAB. The SDPs were solved using SDPT3 [183].

We show the SNR of all reconstructions in the scatter plot of Figure 7.2(a). Here solid dots represent exact rank 1 solutions (largest eigenvalue is at least 100 times larger than the sum of the rest of the eigenvalues), and crosses indicate that the trace minimization found a higher rank solution that has a smaller trace. The percentage of exact rank 1 solutions is shown in Figure 7.2(b). It can be seen that the number of measurements that are needed for perfect recovery is substantially lower than the $13,689$ spikes required by Algorithm 1.



(a) (b)

Figure 7.2: Example of reconstruction by Complex Cell TDM. (a) Effect of number of measurements (spikes) on reconstruction quality. (b) Percentage of rank 1 reconstructions.

7.3 Functional Identification of Complex Cell Energy Processing

Let us now consider a single complex cell with a pair of linear filters $g^1(x,t)$ and $g^2(x,t)$ (dropping the superscript i), whose equivalent second order Volterra kernel is $h_2(x_1,t_1;x_2,t_2)$. In this section, we formulate the functional identification of a complex cell's DSP with the assumption that the DSP is structured according to Figure 7.1(a). We perform M experimental trials. In trial $i, i=1,\cdots,M$, we present to the complex cell the controlled stimulus $u_1^i(x,t)$ and observe the spike times $(t_k^i)_{k\in\mathbb{Z}}$. The objective is to functionally identify h_2 from the knowledge of u^i and $(t_k^i)_{k\in\mathbb{Z}}$, $i=1,\cdots,M$.

7.3.1 Duality Between Functional Identification and Decoding

Definition 16. *Let $h_p \in \mathbb{L}^1(\mathbb{D}^p), p = 1, 2$, where $\mathbb{L}^1$ denotes the space of Lebesgue integrable functions. The operator $\mathcal{P}_1^s : \mathbb{L}_1(\mathbb{D}) \to \mathcal{H}_2^s$ given by*

$$(\mathcal{P}_1^s h_1)(x,t) = \int_{\mathbb{D}} h_1(x',t') K_1(x',t';x,t) dx' dt' \tag{7.16}$$

is called the projection operator from $\mathbb{L}^1(\mathbb{D})$ to $\mathcal{H}_2^s$. Similarly, the operator $\mathcal{P}_2^s : \mathbb{L}_1(\mathbb{D}^2) \to \mathcal{H}_4^s$ given by

$$(\mathcal{P}_2^s h_2)(x_1,t_1;x_2,t_2) = \int_{\mathbb{D}^2} h_2(x_1',t_1';x_2',t_2') K_2(x_1',t_1',x_2',t_2';x_1,t_1,x_2,t_2) dx_1' dt_1' dx_2' dt_2' \tag{7.17}$$

is called the projection operator from $\mathbb{L}^1(\mathbb{D}^2)$ to $\mathcal{H}_4^s$.

Assume that $(\mathcal{P}_1^s g^1)$ and $(\mathcal{P}_1^s g^2)$ are, respectively, of the form

$$(\mathcal{P}_1^s g^m)(x,t) = \sum_{l_x=-L_x}^{L_x} \sum_{l_t=-L_t}^{L_t} c_{l_x l_t}^m e_{l_x l_t}(x,t),$$

where $m = 1, 2$. We have the following

Theorem 8. *The coefficients of a complex cell DSP encoding M space-time test stimuli $u_1^i(x,t), i = 1, \cdots, M$, satisfy the set of equations*

$$\mathbf{Tr}(\mathbf{\Psi}_k^i \mathbf{C}) = q_k^i, k \in \mathbb{I}^i, i = 1, \cdots, M \tag{7.18}$$

where $\mathbb{I}^i, i = 1, \cdots, M$, is the index set of spikes generate by the complex cell in trial i, $\mathbf{C}$ is a rank-2 Hermitian matrix given by $\mathbf{C} = \mathbf{c}^1(\mathbf{c}^1)^{\mathsf{H}} + \mathbf{c}^2(\mathbf{c}^2)^{\mathsf{H}}$, where for $m = 1, 2$,

$$\mathbf{c}^m = \left[c_{-L_x,-L_t}^m, \cdots, c_{-L_x,L_t}^m, c_{-L_x+1,-L_t}^m, \cdots, c_{-L_x+1,L_t}^m, \cdots, c_{L_x,-L_t}^m, \cdots, c_{L_x,L_t}^m\right]^{\mathsf{T}},$$

and $(\mathbf{\Psi}_k^i), k \in \mathbb{I}^i, i = 1, \cdots, M$, are Hermitian matrices with entry at the $(l_{x_2} + L_x + 1 + (l_{t_2} + L_t)(2L_t + 1))$-th row, $(l_{x_1} + L_x + 1 + (l_{t_1} + L_t)(2L_t + 1))$-th column given by

$$\begin{aligned}
&[\mathbf{\Psi}_k^i]_{l_{x_2} l_{t_2}; l_{x_1} l_{t_1}} \\
&= \int_{t_k^i}^{t_{k+1}^i} e_{l_{t_1} - l_{t_2}}(t) dt \int_{\mathbb{D}^2} u_2^i(x_1,s_1;x_2,s_2) e_{l_{x_1},-l_{t_1}}(x_1,s_1) e_{-l_{x_2},l_{t_2}}(x_2,s_2) dx_1 ds_1 dx_2 ds_2.
\end{aligned} \tag{7.19}$$

Proof: Since $u_1^i \in \mathcal{H}_2^s$, we have the following relation [102]

$$\int_{\mathbb{D}} g^m(x, t-s)u_1^i(x,s)dxds = \int_{\mathbb{D}} u_1^i(x, t-s)(\mathcal{P}_1^s g^m)(x,s)dxds.$$

Therefore, the t-transform in (7.7) can be rewritten as $\mathcal{L}_k^i(\mathcal{P}_2^s h_2) = q_k^i, k \in \mathbb{I}^i, i = 1, \cdots, M$, where

$$\mathcal{L}_k^i(\mathcal{P}_2^s h_2) = \int_{t_k^i}^{t_{k+1}^i} \int_{\mathbb{D}^2} u_2^i(x_1, t-s_1; x_2, t-s_2)(\mathcal{P}_2^s h_2)(x_1, s_1; x_2, s_2)dx_1 ds_1 dx_2 ds_2 dt.$$

Since

$$(\mathcal{P}_2^s h_2)(x_1, s_1; x_2, s_2) = (\mathcal{P}_1^s g^1)(x_1, s_1)(\mathcal{P}_1^s g^1)(x_2, s_2) + (\mathcal{P}_1^s g^2)(x_1, s_1)(\mathcal{P}_1^s g^2)(x_2, s_2), \quad (7.20)$$

(7.18) can be obtained following the steps of the proof of Theorem 7. □

Remark 18. *The similarity between* (7.11) *and* (7.19) *by swapping* u_2 *and* h_2 *suggest that the functional identification problem whose objective is to identify the DSP of complex cell by presenting to it multiple stimuli, is dual to the decoding problem whose objective is to reconstruct a stimulus that is encoded by multiple complex cells.*

7.3.2 Complex Cell Channel Identification Machines

To functionally identify the second order Volterra kernel in the complex cell DSP, we again employ a rank minimization problem

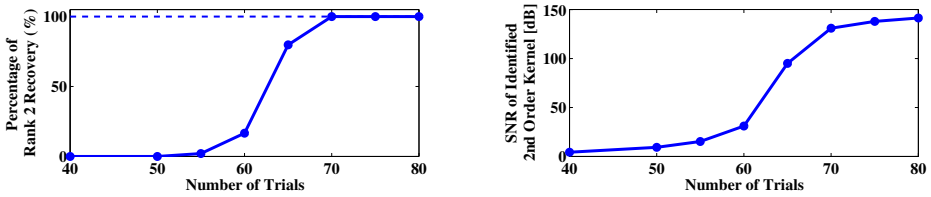
$$\begin{aligned} \text{minimize} \quad & \mathbf{Rank}(\mathbf{C}) \\ \text{s.t.} \quad & \mathbf{Tr}(\mathbf{\Psi}_k^i \mathbf{C}) = q_k^i, k \in \mathbb{I}^i, i = 1, \cdots, M \\ & \mathbf{C} \succcurlyeq 0 \end{aligned} \quad (7.21)$$

Algorithm 2 can similarly be applied. However, in this case, the optimal solution shall have rank 2 and we use the largest two eigenvalues and their respective eigenvectors to identify $\mathcal{P}_2^s h_2$. Note that only the projection of the DSP h_2 onto the space of input stimuli can be identified. Also, note that $\mathcal{P}_1^s g^1$ and $\mathcal{P}_1^s g^2$ can be identified up to a phase parameter since the response of complex cells with quadrature pair Gabor filters is phase-invariant. However the equivalent Hermitian Volterra kernel can be uniquely identified as given in (7.20).

The functional identification of the DSPs of complex cells based on the above algorithm is called Complex Cell Channel Identification Machine (Complex Cell CIM). We evaluate the performance of Complex Cell CIM using simulations in Section 7.3.3.

7.3.3 Examples

First, we considered identifying the DSP of one of the complex cells of the Complex Cell TEM used in Section 7.2.3. We varied the number of experimental trials (and consequently the number of measurements) used in identification. We repeated this process for 200 times. The percentage of rank 2 solutions is shown in Figure 7.3(a) as a function of M, the number of experimental trials. The mean SNR is shown in Figure 7.3(b).



(a) (b)

Figure 7.3: Example of functional identification by Complex Cell CIM. (a) Percentage of successful rank 2 recovery in identification. (b) Mean SNR of identified second order DSP kernel.

Second, we compared the identification performance of the Complex Cell CIM and STC [147] algorithms. Figure 7.4 shows the identified Gabor filters using the Complex Cell CIM and STC algorithms. The number of spikes used are indicated at the top of each column. Note that the Complex Cell CIM reaches perfect identification using only 746 spikes. The pair of Gabor filters used here was orthogonal but the filters had different norms. The different footprint of the individual filters enabled their exact identification.

Third, we evaluated the identification performance in the stimulus space (see also Chapter 5.2). The DSPs of all 62 complex cells of the circuit described in Chapter 7.2.3 were first identified (8 sets, each set identified using a different number of experimental trials). Biologically it is not possible to compare identified DSPs with the underlying DSPs, since the latter are not available. However, instead of comparing responses of neurons with their

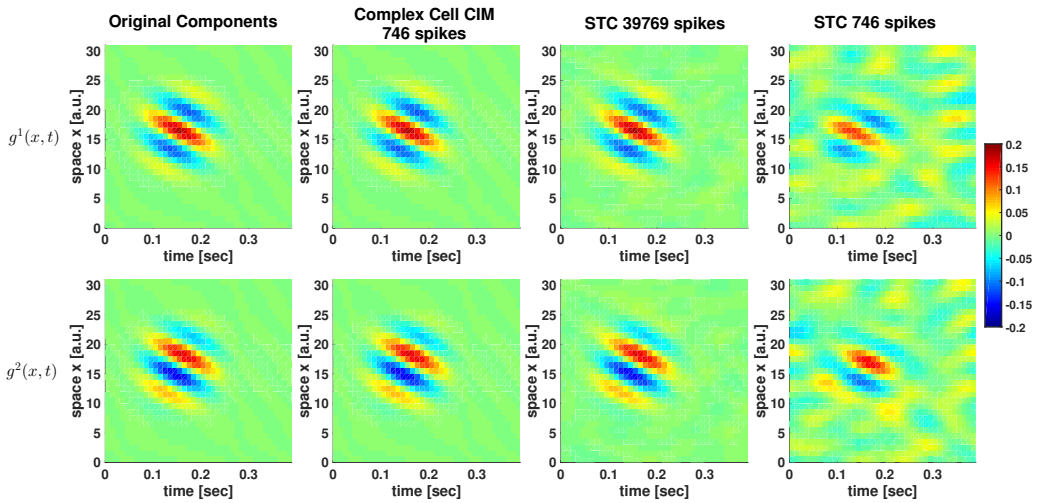


Figure 7.4: Comparison of identification performance by the Complex Cell CIM and STC. Quadrature pair Gabor filters (1st column) identified with Complex Cell CIM with 746 spikes (2nd column, SNR: 123.08 [dB], 88.93 [dB]), and with STC using 39, 769 spikes (3rd column, SNR: 16.77 [dB], 17.90 [dB]) and using 746 spikes (4th column, SNR: 0.21 [dB], 0.52 [dB]).

identified counterparts, we used the decoding algorithm to reconstruct novel stimuli encoded by the underlying Complex Cell TEM assuming that the TEM was instantiated with the identified parameters. Thus, the evaluation of the dual decoding/ identification algorithms can be intuitively performed in the input space. The quality of reconstruction of 100 stimuli based on the 8 sets of identified DSPs is depicted in Figure 7.5 using the mean SNR. The results suggest that the identification of the entire neural circuit is accurate after 70 experimental trials.

7.4 Extensions

We noted that the dimensionality of higher order Volterra DSPs pose a strong practical limitation on their identification. For computationally tractable identification, it is therefore imperative to impose additional constraints on the higher order Volterra DSPs. The energy model of the complex cells we discussed here is a particular case of low-rank second order

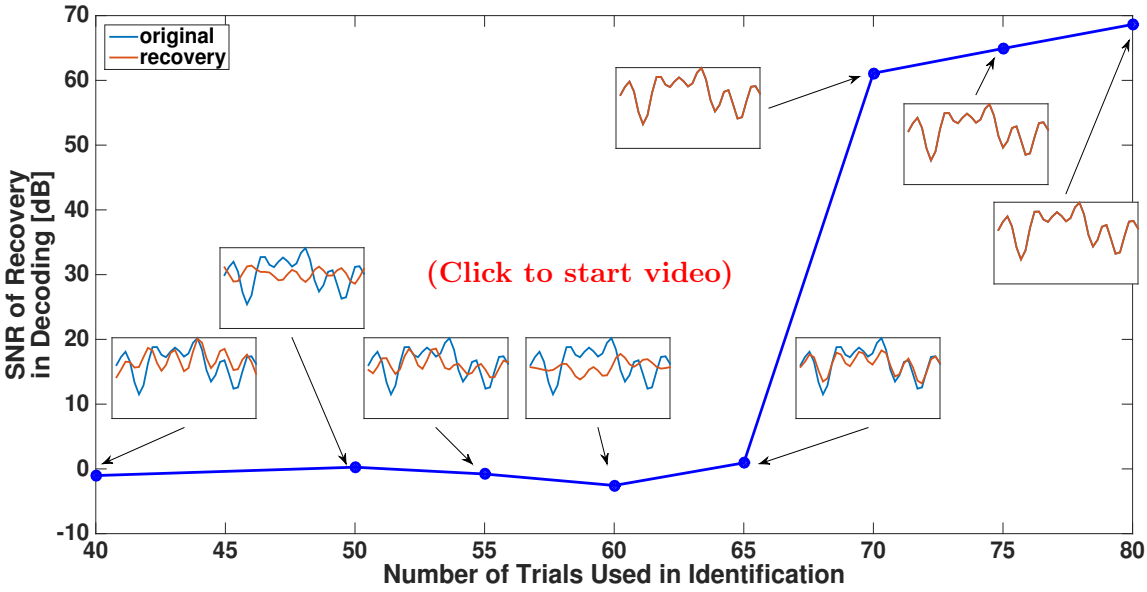


Figure 7.5: Evaluating identification quality of the Complex Cell CIM in the input space. SNR of reconstruction of novel stimuli assumed to be encoded with the identified DSPs.

Volterra DSPs.

Instead of the sum of squares of individual linear filters, h_2 can also take the form of

$$h_2(x_1,t_1;x_2,t_2) = \sum_{n=1}^{N} g^{1n}(x_1,t_1)g^{2n}(x_2,t_2), \tag{7.22}$$

where $N \ll dim(\mathcal{H}_2^s)$. DSPs of complex cells can be viewed as a special case of the above model when $N = 2$ and $g^{1n} = g^{2n}, n = 1,2$. Such models also arise naturally in gain control [59] and Reichardt motion detectors [15]. Identification of the second order Volterra DSP above can be obtained with the rank minimization algorithm with minor modifications. The number of measurements and experimental trials required for perfect identification are expected to increase as N increases.

We also expect that the results we obtained here can be generalized to 3-dimensional space-time visual stimuli. An example of reconstruction of a 3-dimensional space-time visual stimuli is shown in Figure 7.6. In this example, an 400-millisecond-long artificial visual stimulus $u(x,y,t)$ was encoded by 318 complex cells. They produced a total of 1374 spikes. The order of the RKHS is $L_x = L_y = 3, L_t = 4$, Therefore, dimension of the stimulus space

is 441. Using the Complex Cell TDM, the visual stimulus was perfectly reconstructed with an SNR of 95.81 [dB]. Note that the number of spikes used to reconstruct the stimulus was about 3 times the dimension of the stimulus. By comparison, perfect reconstruction of this stimulus requires $100,000$ spikes and about $10,000$ neurons if the algorithm 1 is used instead.

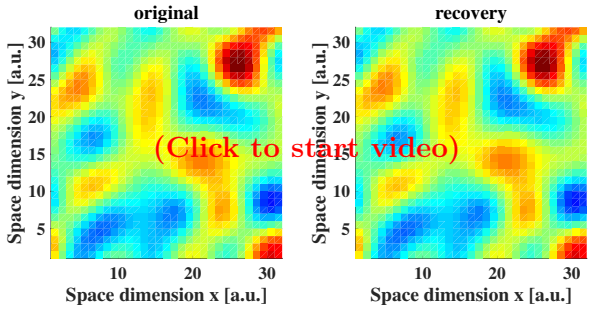


Figure 7.6: Example of Complex Cell TDM of full visual stimulus $u(x, y, t)$. The original video encoded is shown on the left. The stimulus was encoded by 318 complex cells. The reconstructed stimulus using 1374 spikes is shown on the right (SNR=95.81 [dB]). The dimension of the RKHS is 441.

7.5 Summary

In this chapter, we constructed a Complex Cell Video TEM consisting of a population of M complex cells that encode space-time stimuli. We demonstrated that complex cells exhibit Volterra dendritic stimulus processors (Volterra DSPs) that are analytically and computationally tractable. Decoding and identification problems arising in neural circuits built with complex cells can be efficiently solved as rank minimization problems.

We provided a Complex Cell Video TDM that reconstructs visual stimuli encoded by the Complex Cell Video TEM. The decoding algorithm does not suffer from the curse of dimensionality. As a result, it can be applied to decode visual stimuli encoded in a massively parallel fashion. The result we obtained here also suggests that although phase information

is lost in the energy processing in the complex cells, an ensemble of complex cells can still faithfully represent visual stimuli. This result is unexpected from our previous understanding of the roles of complex cells [22].

Furthermore, by duality, we proposed a Complex Cell CIM, enabling functional identification of the energy processing in complex cells using the spike times they generate. We showed that the identification algorithm substantially outperforms the STC in terms of both accuracy and the number of spikes required. Finally, the quality of functional identification by the Complex Cell CIM was evaluated in the stimulus space, as discussed in Chapter 5.

Chapter 8

Identity Preserving Transformation on Visual Scenes in the Spike Domain

The brain must be capable of forming object representations that are invariant with respect to the large number of fluctuations occurring on the retina [41]. These include object position, scale, pose and illumination, and the presence of clutter. In a simple model of the visual system in primates, the incoming visual stimulus is first represented in the responses of the retinal ganglion cells (RGC). Subsequently, the stimulus is re-represented at each neural layer starting with the first relay center (LGN) and followed by the visual cortex (V1, V2, V4 and IT cortex). Each of these representations can be modeled as an Identity Preserving Transformation (IPT). At the final stage, the visual objects are represented in a way that is amenable to an efficient comparison with an internal (memory) representation of the object. Since spike trains are the language of the brain, the latter representation is in the form of a neural population activity. Consequently, the decision whether the object is present or absent takes place in the spike domain [123].

What are some plausible computational or neural mechanisms by which invariance could be achieved? An early pioneering work [141] provides a model mechanism for shifting and rescaling the representation of an object from its retinal reference frame into an object-centered reference frame (see also [4]). In one class of models used in the invariant recognition

literature, transformations of the incoming visual signal are matched with an existing stored version of the image [16]. More formally, let u be a visual sensory object (stimulus). An IPT acting on u is modeled as an invertible transformation $\mathcal{T}$ that, in turn, consists of a composition of a set of elementary operators (e.g., rotation, dilation, translation, etc.). The set of all spike trains produced by $\mathcal{T}(u)$ for all possible IPTs $\mathcal{T}$ defines the object-manifold. For identifying the instantiation of a stored object in the incoming object-manifold, the algorithm presented in [6] calls for the identification of the operator $\mathcal{T}$ (and its inverse). More recent research focuses on routing/connectivity operators in support of information delivery (e.g., sensory information) to higher brain centers [199].

In this chapter we focus on the *realization* of IPTs in the *spike domain*. The spike domain is a non-linear, stimulus-dependent representation space. The non-linear nature of the stimulus representation has proven to be a major challenge for spike domain computation. Our goal here is to put forth the first efficient rigorous computational model that allows formal reasoning in the spike domain while at the same time it is biologically relevant. Our model of computation can briefly be summarized in block diagram form in Figure 8.1. The input visual stimulus is encoded in the time domain by an instantiation of a Video Time Encoding Machine (Video TEM) [98]. The encoded stimulus (in the form of spikes) is first processed in the Time Domain Processing (TDP) block and then decoded by a Time Decoding Machine (TDM). The output of the TDM is again an analog signal. Our time (spike) domain computation chain resembles the traditional digital signal processing [142] chain where an analog signal is converted into a digital signal using an analog-to-digital converter, then processed with a digital signal processor and finally converted back to an analog signal with a digital-to-analog converter.

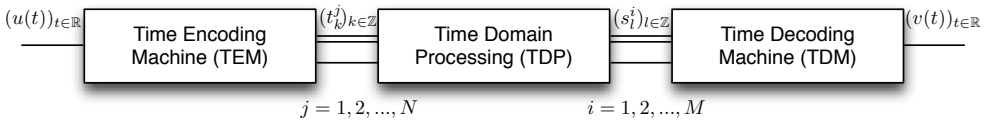


Figure 8.1: General Signal Processing Chain with a Time Domain Core. (Modified from [114] with permission, ©2013 Elsevier)

An example of processing in the time domain appeared in [94] where it was demon-

strated how an arbitrary linear filter can be implemented in the time domain, using neural components. By building upon these results, any IPT acting on the input stimulus can be realized in the time domain. However, the setup of [94] is rather complex as it requires a different TDM for any desired transformation of the sensory stimulus.

There are two types of operators that are used for encoding of stimuli with TEMs: linear operators (receptive fields) followed by non-linear operators (spiking neural circuits). These operators are cascaded. The efficient realizability of IPTs presented here is primarily due to the structure of the receptive fields of the Video TEM. These are required to form an overcomplete spatial (or spatiotemporal) filterbank. Furthermore the set of receptive fields has to exhibit certain symmetry properties (in group theoretic sense). If the receptive fields (linear filters) have a group structure transformations on the stimulus can be realized via transformations on the filters. However, these group operations cannot be, in general, "propagated" through the neural encoding circuits (formally non-linear operators). Surprisingly, however, under certain conditions described in the chapter, rotations, scaling and translations can be efficiently executed in the spike domain.

We show that a large class of IPTs can be efficiently realized by making connectivity changes in the TDP block while the TDM block remains the same. Proposed originally in [151], the TDP block consists of a "switching matrix" that simply regulates the connectivity between the TEM and TDM blocks. We will show that different IPTs can be realized with different connectivity settings of the switching matrix. For example, for a particular setting of the switching matrix, the original stimulus is faithfully recovered. For other settings, translations, rotations and dilations (or combinations of these transformations) of the original video stream are obtained. We will also show that IPTs can be computed in parallel. Our model can be viewed as a generalization of the shifting and rescaling mechanisms proposed in [141]. We extend these operations to include rotations and show how to efficiently implement them in the spike domain.

A key difference in the study in this chapter from that in [151] is that we take into account the constraints that the finite size of the neural population imposes on the set of achievable transformations. By starting from the continuous group on the plane characterizing all the possible IPTs, we advance two different encoding architectures whose receptive fields are

defined on two different discrete grids. The first is a log-polar grid, similar to the ones used in models of foveated vision [137, 193, 198]. On the log-polar grid the switching matrix can realize combinations of rotations and dilations in a lossless manner in the spike domain. The second is a Cartesian grid [55, 120]. On the latter grid the switching matrix can realize combinations of dilations and translations in a lossless manner in the spike domain as well. Finally, we discuss how discrete approximations of the continuous symmetry group can be used to perform arbitrary but approximate IPTs in the spike domain. Examples are given that intuitively demonstrate our methodology.

8.1 Processing of Visual Stream in the Spike Domain

8.1.1 The Architecture of the Model of Computation

An illustration of a general switching ("rewiring") architecture for encoding, processing and decoding video streams is shown in Figure 8.2. Our architecture follows the general one depicted in Figure 8.1 and in [151].

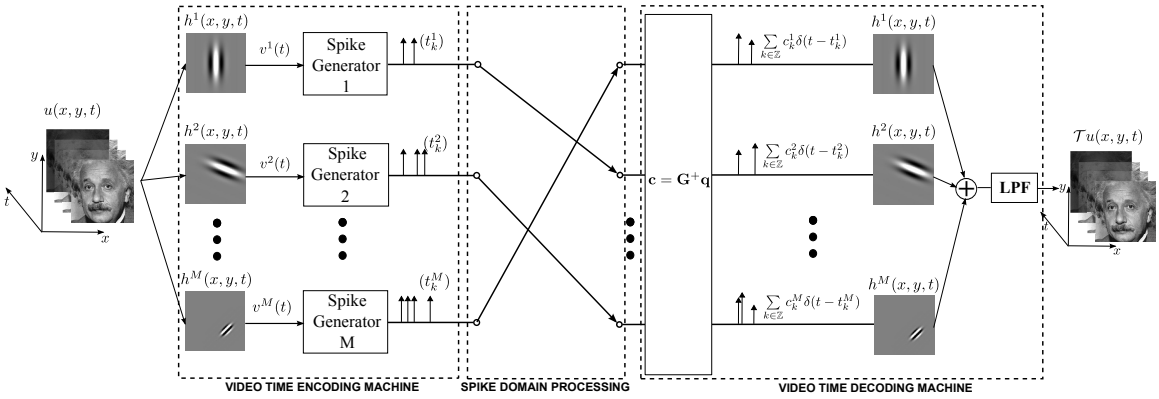


Figure 8.2: General architecture of the video processing mechanism using spike domain switching techniques. (Modified from [114] with permission, ©2013 Elsevier)

The input signal is an analog video stream and is encoded by a Video Time Encoding Machine (see Figure 8.2). The Video TEM consists of a bank of linear DSPs/receptive fields $h_1^i(x,y,t), i = 1 \cdots, M$, in cascade with nonlinear spiking circuits (e.g., neural circuits realized with Integrate-and-Fire neurons).

The TDM block implements decoding algorithms for the Video TEM (see Figure 8.2). Under certain conditions the Video TEMs can faithfully encode the input video stream as a multidimensional sequence of spike trains. The TDM architecture implements a perfect decoding algorithm of the input video stream (see Section 3.3.1). For the rest of this chapter we assume that the number and the parameters of the neurons are such that the perfect stimulus recovery conditions are satisfied.

The architecture of the TDP block of Figure 8.2 is very simple. It consists of a *switching matrix* that regulates the connectivity between the TEM and TDM blocks [151]. In other words, the switching matrix, directs the incoming spikes from the layer of neural circuits that represents the incoming video stimulus, to specific locations of the next layer, i.e., the TDM block or other layers of read-out neurons. For a finite number of M circuits, the switching matrix can have $M!$ different settings. Each setting corresponds to a permutation σ of the numbers $\{1, 2, \ldots, M\}$ mapping the spikes coming from the neural circuit i to the $\sigma(i)$-th entry of the next block or layer. Clearly such a transformation, although non-linear in general, is identity preserving in the time domain because it is invertible through the permutation σ^{-1}. However not all of these transformations have a clear physical interpretation. Moreover, the representations in the visual system have to be invariant with respect to certain transformations. These include rotation, dilation (scaling) and translation among others. In what follows we show how the structure of the Video TEM together with the operation of the switching matrix can give rise to such invariant representations.

8.1.2 Identity-Preserving Transformations in the Spike Domain

In this section we present the general architecture for the realization of Identity-Preserving Transformations (IPTs) in the spike domain by means of switching mechanisms that regulate connectivity. We argue that IPTs naturally arise in the time domain representation when using a Video TEM provided that the set of receptive fields has some special symmetry properties (Sections 8.1.2.1 and 8.1.2.2). We present two different sets of receptive fields that can realize various IPTs in an exact form (Sections 8.1.2.3 and 8.1.2.4) and also describe the realization of approximate arbitrary IPTs by our architecture (Section 8.1.2.5). Finally, we assume that all spike generation model neurons have the same parameters (i.e., same

threshold, bias and time constants for the case of Integrate-And-Fire (IAF) neurons and the same feedback loop if any).

8.1.2.1 The Structure of Receptive Fields

We consider receptive fields that are space-time separable, i.e., they satisfy

$$h_1^i(x, y, t) = {}^s h_1^i(x, y) \cdot {}^\tau h_1^i \tau(t) \tag{8.1}$$

for all $i, i = 1, 2, \ldots, M$. Moreover, we assume that the temporal component ${}^\tau h_1^i$ is the same for all the receptive fields, and that its spectral support covers the frequency band of interest $[-\Omega, \Omega]$. Thus, there is no information loss due to temporal filtering.

To generate a set of spatial receptive fields, we pick a *mother* function $\eta \in L^2(\mathbb{R}^2)$, similar to the mother wavelet in wavelet theory [35]. Then, each individual receptive field is obtained by applying to η the unitary operator

$$\mathcal{T}([x_0, y_0], \alpha, \theta)\eta(x, y) = \tau_{x_0,y_0} D_\alpha R_\theta \eta(x, y), \tag{8.2}$$

where

1. $\tau_{x_0,y_0}, (x_0, y_0) \in \mathbb{R}^2$ with $\tau_{(x_0,y_0)}\eta(x, y) = \eta(x - x_0, y - y_0)$ is the translation operator;
2. $\mathcal{D}_\alpha, \alpha > 0$ with $\mathcal{D}_\alpha \eta(x, y) = \alpha^{-1}\eta\left(\frac{x}{\alpha}, \frac{y}{\alpha}\right)$ is the dilation operator;
3. $R_\theta, \theta \in [0, 2\pi)$, with $R_\theta\eta(x, y) = \eta(r_{-\theta}[x, y])$, where $r_\theta[x, y] = [x\cos\theta - y\sin\theta, x\sin\theta + y\cos\theta]$ is the rotation operator.

We *define* the operator of (8.2) to be equal to

$$\mathcal{T}([x_0, y_0], \alpha, \theta)\eta(x, y) = \alpha^{-1}\eta(\alpha^{-1} r_{-\theta}(x - x_0, y - y_0)). \tag{8.3}$$

In fact, the family of the operators $\mathcal{T}([x_0, y_0], \alpha, \theta)$ is the unique (up to unitary equivalence) unitary irreducible representation of a group called the *similitude group* SIM(2) [5].

Adapting the group notation, we define an element of SIM(2) $g = ([x_0, y_0], \alpha, \theta)$, where $x_0, y_0 \in \mathbb{R}, \alpha \in \mathbb{R}^+$ and $\theta \in [0, 2\pi)$. Each of $g \in$ SIM(2) is an elementary transformations (translations, dilations and rotations) on the $\mathbb{R}^2$ plane, where the transformation is given by

$$[x', y'] = ([x_0, y_0], \alpha, \theta)[x, y] = \alpha r_\theta[x, y] + [x_0, y_0].$$

The group law (which consists by the operation $\circ$ between two elements, the identity element, and the inverse element) is given by

$$\begin{aligned} g' \circ g = ([x_0', y_0'], \alpha', \theta') \circ ([x_0, y_0], \alpha, \theta) = ([x_0', y_0'] + \alpha' r_{\theta'}[x_0, y_0], \alpha'\alpha, \theta' + \theta), \\ e = ([0, 0], 1, 0), \\ g^{-1} = ([x_0, y_0], \alpha, \theta)^{-1} = (-\alpha^{-1} r_{-\theta}[x_0, y_0], \alpha^{-1}, -\theta), \end{aligned} \tag{8.4}$$

and the associativity can be easily verified.

By denoting the receptive field $\eta_g, g = ([x_0, y_0], \alpha, \theta)$, where

$$\eta_g(x, y) = \mathcal{T}(g)\eta(x, y), \tag{8.5}$$

we notice that the action of another operator $\mathcal{T}(g')$ on η_g will result in

$$\mathcal{T}(g')\eta_g = \eta_{g' \circ g}, \tag{8.6}$$

where the subscript of the resulting receptive field is given by the group law. With the above receptive field structure, we present next the generation of invariant transformations that are based on a switching mechanism.

8.1.2.2 Generation of Invariant Transformations

Let us denote by S a subset of the SIM(2) group, and assume that a signal of interest $A(x, y)$ (for simplicity we assume a constant image, although this approach is readily applicable to time-varying signals as well) can be represented as

$$A(x, y) = \sum_{s \in S} c_s \eta_s(x, y). \tag{8.7}$$

To apply a transformation $\mathcal{T}(g), g \in H \subseteq SIM(2)$, to A, we have

$$\begin{aligned} \mathcal{T}(g)A(x, y) &= \sum_{s \in S} c_s \mathcal{T}(g)\eta_s(x, y) \\ &= \sum_{s \in S} c_s \eta_{g \circ s}(x, y) \end{aligned} \tag{8.8}$$

If, in addition, $g \circ s \in S$ and $g^{-1} \circ s \in S$ for all $s \in S$, then the subset S is invariant under the action of $\mathcal{T}(g)$. Consequently, the transformation of A by $\mathcal{T}(g)$ can be rewritten as

$$\mathcal{T}(g)A(x, y) = \sum_{s \in S} c_{g^{-1} \circ s} \eta_s(x, y) \tag{8.9}$$

Therefore, we see that the transformed image $\mathcal{T}(g)A$ takes the same representation as A itself. The only changes are the coefficients $c_s, s \in S$. As indicated in (8.9), the coefficient of the receptive field η_s in $\mathcal{T}(g)A$, should be the coefficient of the receptive field $\eta_{g^{-1}\circ s}$ in A, for all $s \in S$. This is what the switching architecture of the spike domain process requires, *i.e.*, any IPT on the set of receptive fields maps to the same set of receptive fields. Once we know the representation of (8.7), the transformation can be easily realized with a switching circuit.

The invariance of the set S under the action of any $\mathcal{T}(g), g \in H$, is obvious if $H = S = SIM(2)$. In essence, the closure property of the group operation guarantees that the transformation of a receptive field by the unitary operator is another receptive field in the group. It is certainly true that under the continuous group structure, if there are uncountably many receptive fields $\eta_{([x_0,y_0],\alpha,\theta)}(x, y)$, for all $[x_0, y_0] \in \mathbb{R}^2, \alpha \in \mathbb{R}^*_+, \theta \in [0, 2\pi)$, then all rotations, translations and dilations can be implemented by a switching mechanism.

However, there is only a finite number of receptive fields since there is a finite number of neurons. Theoretically, we can make this number countably infinite, but it still requires the discretization of the SIM(2) group, and the restriction of all possible operations to a discrete set. Unfortunately, discretization breaks the nice properties of the continuous group and specific designs of the discretization and switching mechanisms are necessary. We present here a discretization based on a log-polar grid that is invariant under the action of a discrete set of rotations and dilations, and discuss the transformations that it can realize. In Section 8.1.2.4 we present an alternative discretization that is based on a Cartesian grid and is invariant under the action of a discrete set of translations and dilations.

8.1.2.3 Exact IPTs Using a Log-Polar Grid

In the case of the log-polar grid, the centers of the receptive fields are placed in a rotation-invariant grid. More specifically, the discretization of the SIM(2) group is given by the

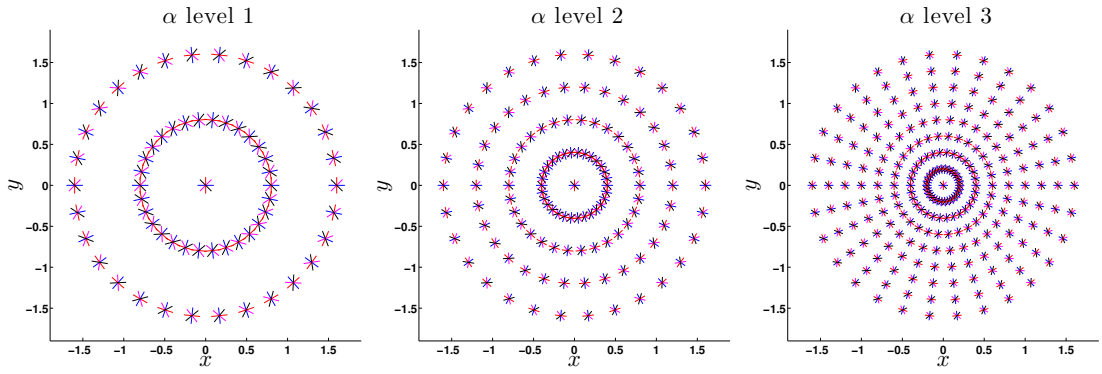


Figure 8.3: An example of discretization of the SIM(2) group to Log-Polar Grid, as defined by (8.10), where $N = 8, L = 30, \alpha_0 = 2, b_0 = 0.8$, and the spatial grid of three levels of $m \in \{0, -1, -2\}$ is depicted in each of the subfigures, corresponding to three different dilation levels (dilation parameter of the group elements). Each element in the grid is represented by a small oriented line segment. The spatial position of the center of a line segment corresponds to the translation parameter of group element, and the angle between a line segment and x axis corresponds to the rotation parameter of the group element. The color of each line segment corresponds to a particular value of n in (8.10) modulo 4, where blue, magenta, red and black correspond to $n = 0, 1, 2, 3$, respectively. Note that at the origin, only one of the L elements are shown for clarity.

subset (see Figure 8.3)

$$S_p = \left\{ (\alpha_0^m r_{l\theta_0}[kb_0, 0], \alpha_0^m, n\omega_0 + l\theta_0) \middle| \begin{array}{l} b_0 > 0, k \in \mathbb{N}, m \in \mathbb{Z}, \alpha_0 > 1 \\ \omega_0 = 2\pi/N, N \in \mathbb{N}^*, n = 0, \cdots, N-1 \\ \theta_0 = 2\pi/L, L \in \mathbb{N}^*, l = 0, \cdots, L-1 \end{array} \right\}. \quad (8.10)$$

Having (8.3) in mind, we have that the general form of the receptive field constructed according to S_p is given by

$$\eta_{(\alpha_0^m r_{l\theta_0}[kb_0,0], \alpha_0^m, n\omega_0 + l\theta_0)} = \alpha_0^{-m} \eta(\alpha_0^{-m} r_{-(n\omega_0 + l\theta_0)}([x, y] - \alpha_0^m r_{l\theta_0}[kb_0, 0])). \quad (8.11)$$

We see that for each scale α_0^m, this set contains receptive fields that are centered in the points $\alpha_0^m r_{l\theta_0}[kb_0, 0]$, and have orientation $n\omega_0 + l\theta_0$. Note that the term $n\omega_0$ corresponds

to the local orientation of the receptive fields around their center point. In the case where the receptive fields are isotropic, we can simply set $N = 1$. On the contrary, the term $l\theta_0$ corresponds to the global orientation, i.e., the angle between the line that connects the origin and the center of the receptive field and the x-axis. Since elements of S_p are uniquely determined by the parameters k, m, n and l, we use a more compact notation (k, m, n, l) to denote elements in S_p. We would like that the subset is invariant under some rotations and dilations. From the discretization of the SIM(2) group, it naturally arises that those rotations and dilations are derived from the following subset of SIM(2)

$$H_p = \left\{([0,0], \alpha_0^m, l\theta_0) \middle| m \in \mathbb{Z}, \alpha_0 > 1, \theta_0 = 2\pi/L, L \in \mathbb{N}^*, n = 0, \cdots, L-1\right\}. \tag{8.12}$$

Proposition 1. *For every choice of the parameters $(b_0, \alpha_0, \omega_0, \theta_0)$ with $b_0 > 0, \alpha_0 > 1, N, L \in \mathbb{N}^*$ of the log-polar grid, the set of all constructed receptive fields, denoted by S_p, is invariant under any transformations in the subset H_p. In addition, the action of each element of H_p induces a permutation of S_p.*

Proof: The proposition is a direct consequence from the way the elements of S_p and H_p are constructed. Indeed, for each $h = ([0,0], \alpha_0^{m'}, l'\theta_0) \in H_p$ and $x = (\alpha_0^m r_{l\theta_0}[kb_0, 0], \alpha_0^m, n\omega_0 + l\theta_0) \in S_p$, we have

$$h \circ x = (\alpha_0^{m'+m} r_{(l'+l)\theta_0}[kb_0, 0], \alpha_0^{m'+m}, n\omega_0 + (l'+l)\theta_0) \in S_p. \tag{8.13}$$

Note that H_p is a subgroup of SIM(2) under the same action, since the identity element $([0,0], 1, 0) \in H_p$ and for each $h = ([0,0], \alpha_0^{m'}, l'\theta_0) \in H_p$ we have $h^{-1} = ([0,0], \alpha_0^{-m'}, (-l' \mod L)\theta_0) \in H_p$. Therefore, it is easy to see that for each $x_1, x_2 \in S_p$, if $hx_1 = hx_2$, then $h^{-1}hx_1 = h^{-1}hx_2$ and thus $x_1 = x_2$. Therefore, the mapping of S_p to itself by h is one-to-one and $h \in H$ induces a permutation of S_p.

Before we present the main result for this section we need the following definitions.

Definition 17. *A set $\mathbb{T} := \left\{(t_k^i), k \in \mathbb{Z}, j = 1, 2, \ldots, M\right\}$ of spike trains produced by the TEM of Figure 8.2 is said to represent the video stimulus u if the TDM of Figure 8.2 with given input the set of spike trains $(t_k^i), k \in \mathbb{Z}, i = 1, 2, \ldots, M$ recovers the video stream.*

Definition 18. *Let u be an arbitrary input video stream and $\mathbb{T}$ the set of spike trains produced by the TEM of Figure 8.2. An IPT $\mathcal{T}$ is said to be realizable in the time domain, if there is*

a connectivity setting σ of the switching matrix, such that the set of spike trains $\mathbb{T}$ represents the video stimulus $\mathcal{T}u$.

Theorem 9. *For a Video TEM with receptive fields according to S_p the following IPTs are realizable in the time domain:*

- *The set of rotations $\mathcal{R}_\theta$ where $\theta = l\frac{2\pi}{L}, l = 0, 1, \ldots, L-1$.*
- *The set of dilations $\mathcal{D}_\alpha$ where $\alpha = \alpha_0^m, m \in \mathbb{Z}$.*
- *Any synthesis of the above operators.*

Proof: See Section B.6 for a proof.

Definition 19. *Let u denote the video stream that represents a certain object and let $\mathbb{T}$ be the set of spike trains obtained from the Video TEM upon presentation of u. The set $\{\mathcal{T}u : \mathcal{T} \in H\}$ is called the object manifold under H for the object that is represented with u.*

Corollary 1. *The switching architecture of Figure 8.2 where the set of receptive fields is characterized by S_p generates the whole object manifold under H_p in real time.*

Proof: Follows directly from the above proof of Theorem 9.

Remark 19. *The theory presented above considers video streams with spatial support on $\mathbb{R}^2$. Moreover, an implicit requirement of Theorem 9 is that the number of elements in the set of receptive fields (which equals the number of spiking neural circuits) is infinite. In applications however, as well as in the visual system, the number of neurons is finite. This means that the number of possible scalings as well as the number of possible translations is finite. The finite number of possible translations implies that the spatial domain over which the input video stream is defined is of finite measure, that is clearly the case. The finite number of scalings implies that the input stimuli have also finite spatial bandwidth which also holds. These facts also restrict the set of possible IPTs that can be implemented in the time domain, to those that are supported by the characteristics of the set of receptive fields. Note however that the set of possible rotations remains unaffected.*

8.1.2.4 Exact IPTs using a Cartesian Grid

From the presentation of the log-polar grid in Section 8.1.2.3, we see that rotation and dilation transformations can be realized in an exact fashion by the same switching circuit. This is because these transformations commute with each other, and thus it is possible to derive a discrete set that preserves the group structure. On the contrary, the translation transformation does not commute neither with rotations nor with dilations. As a result, it is impossible to derive a general discrete set that will be closed under *all* the combined transformations of translation and rotation or dilation. However, we present here a discrete set that can realize a subset of these transformations on the Cartesian grid, in an exact fashion.

In this case the centers of the receptive fields for each scale are placed on a Cartesian grid. Spatial filters of multiple scales are constructed, with the scales placed logarithmically. The resulting discretization of the SIM(2) group is given by (see Figure 8.4)

$$S_c = \left\{ (\alpha_0^{-m}[kb_x, lb_y], \alpha_0^{-m}, n\omega_0) \middle| \begin{array}{l} b_x, b_y > 0, k, l \in \mathbb{Z} \\ \alpha_0 \in \mathbb{N}^*, \alpha_0 > 1, m \in \mathbb{N} \\ \omega_0 = 2\pi/N, N \in \mathbb{N}^*, n = 0, \cdots, N-1 \end{array} \right\}. \tag{8.14}$$

Again from (8.3), the general receptive field obtained from the above discretization is given by

$$\eta_{(\alpha_0^{-m}[kb_x, lb_y], \alpha_0^{-m}, n\omega_0)} = \alpha_0^m \eta(\alpha_0^m r_{-n\omega_0}([x, y] - \alpha_0^{-m}[kb_x, lb_y])). \tag{8.15}$$

For each scale α_0^{-m} the set includes the receptive fields centered at $\alpha_0^{-m}[kb_x, lb_y]$, with orientation $n\omega_0$. Note that this discretization resembles closely the one of the discrete wavelet transform. We would like S_c to be invariant under some specific translations and dilations. Based on the above discretization, this set of transformations is given by

$$H_c = \left\{ (\alpha_0^{-m'}[k'b_x, l'b_y], \alpha_0^{-m'}, 0) | m' \in \mathbb{N}, k', l' \in \mathbb{Z} \right\}. \tag{8.16}$$

As in the case of the log-polar grid, consider an element $h = (\alpha_0^{-m'}[k'b_x, l'b_y], \alpha_0^{-m'}, 0) \in H_c$ and an element $x = (\alpha_0^{-m}[kb_x, lb_y], \alpha_0^{-m}, n\omega_0) \in S_c$. Then we have

$$\begin{aligned} h \circ x &= (\alpha_0^{-m'}[k'b_x, l'b_y] + \alpha_0^{-m'-m}[kb_x, lb_y], \alpha_0^{-(m+m')}, n\omega_0) \\ &= (\alpha_0^{-(m+m')}[(\alpha_0^m k' + k)b_x, (\alpha_0^m l' + l)b_y], \alpha_0^{-(m+m')}, n\omega_0), \end{aligned} \tag{8.17}$$

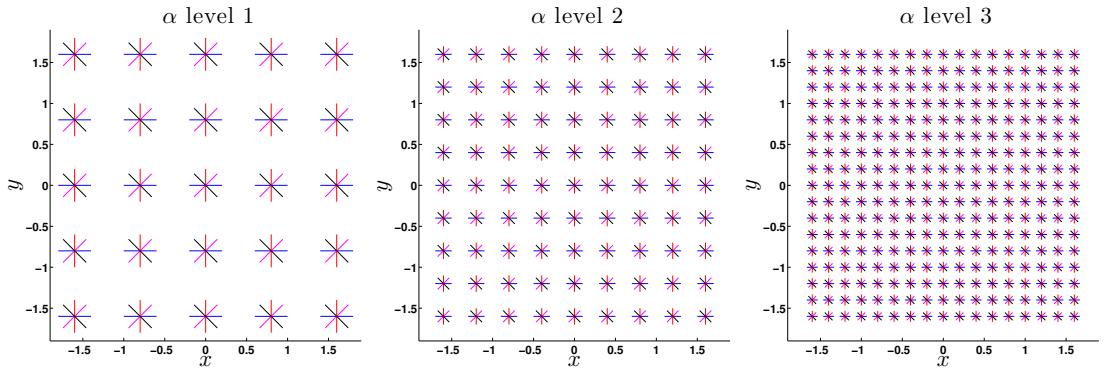


Figure 8.4: An example of discretization of the SIM(2) group to Cartesian Grid, as defined by (8.14), where $N = 8, \alpha_0 = 2, b_0 = 0.8$, and the spatial grid of three values of $m \in \{0, 1, 2\}$ is depicted in each of the subfigures, corresponding to three different dilation levels (dilation parameter of the group elements). Each element in the grid is represented by a small oriented line segment. The spatial position of the center of a line segment corresponds to the translation parameter of group element, and the angle between a line segment and x axis corresponds to the rotation parameter of the group element. The color of each line segment corresponds to a particular value of n (8.14) modulo 4, where blue, magenta, red and black correspond to $n = 0, 1, 2, 3$, respectively.

and since α_0 is a positive integer, we have that $\alpha_0^m k' + k, \alpha_0^m l' + l \in \mathbb{Z}$ and therefore $h \circ x \in S_p$.

Based on the above discussion, we have the following theorem for the IPTs that are implementable with a Cartesian grid.

Theorem 10. *For a Video TEM with receptive fields according to S_c the following IPTs are realizable in the time domain:*

- *The set of dilations $\mathcal{D}_\alpha$ where $\alpha = \alpha_0^{-m}, m \in \mathbb{N}$.*
- *The set of translations $\tau_{[x_0,y_0]}$ where $[x_0, y_0] = [kb_x, lb_y], k, l \in \mathbb{Z}$.*
- *Any synthesis of the above operators, i.e., simultaneous dilation $\mathcal{D}_{\alpha_0^{-m}}$ and translation $\tau_{\alpha_0^{-m}[kb_x, lb_y]}, m \in \mathbb{N}, k, l \in \mathbb{Z}$.*

Proof: Similar to the one of Theorem. 9. □

Remark 20. *Note that the only dilations that the Cartesian grid can implement are with a scale $\alpha_0^{m'}$, where m' is a strictly positive integer, and not a general integer as in the log-polar grid case. This corresponds only to the zoom-out transformation. The opposite zoom-in transformation cannot be performed in an exact way since for $m' < 0$, we also have $m + m' < 0$ for $m = 0, \ldots, -m' - 1$. However, note that even when this condition is not satisfied, we can perform approximate zoom-in transformations by keeping only the scales $-m', -m' + 1, \ldots$ and disregarding the remaining ones. We present such an example in Section 8.2.2.*

8.1.2.5 Approximate IPTs Using Nearest Neighbor Mapping

As we discussed, the two different grids, presented in sections 8.1.2.3 and 8.1.2.4 correspond to two different discretizations of the continuous SIM(2) group that consists of all possible rotations, translations and dilations. Conceptually, the SIM(2) group can generate all the possible IPTs and therefore has led in the past to the development of group theoretic approaches for the problem of visual perception [42, 78]. However, these ideas cannot lead to practical applications since they require an uncountable number of elements and the general group structure is not retained for arbitrary discretization schemes. Nevertheless, both of the grids can approximate the continuous one as they become denser, i.e., the distance between neighboring elements becomes smaller. The polar grid becomes dense in the continuous group as $N \to \infty$, $\alpha_0 \to 1^+$ and $b_0 \to 0^+$. Similarly, the Cartesian grid becomes dense in the continuous group as $b_x, b_y \to 0$, $\alpha_0 \to 1^+$ and $N \to \infty$. Therefore as the grids become denser, they also become more similar to the continuous one.

Using this argument we can also use a Video TEM with a set of receptive fields placed on a Cartesian grid to implement approximate rotations, of an arbitrary angle θ. To do so we set the switching matrix as follows: Upon the application of a rotation operator r_θ the spikes of the neural circuit with receptive field that corresponds to the point $(\alpha_0^{-m}[kb_x, lb_y], \alpha_0^{-m}, n\omega_0)$ is mapped, to its nearest neighbor with the same scale $(\alpha_0^{-m}[k'b_x, l'b_y], \alpha_0^{-m}, n'\omega_0)$ where

k', l', n' are given by

$$\begin{aligned}
[k', l'] =& \underset{i,j\in\mathbb{Z}}{\text{argmin}} \left\{(kb_x\cos\theta + lb_y\sin\theta - ib_x)^2 + (-kb_x\sin\theta + lb_y\cos\theta - jb_y)^2\right\} \\
=& \left[\text{nint}(k\cos\theta + lb_y\sin\theta/b_x), \text{nint}(-kb_x\sin\theta/b_y + l\cos\theta)\right], \\
n' =& \underset{l\in\mathbb{Z}/N}{\text{argmin}} |n\omega_0 + \theta - l\omega_0|
\end{aligned} \tag{8.18}$$

where $\text{nint}(x)$ is the nearest integer to x. Note that the distance to the closest receptive field is always bounded by $\sqrt{b_x^2 + b_y^2}/2$ which ensures that as b_x, b_y becomes smaller, i.e., the grid becomes denser, the mapping becomes more accurate.

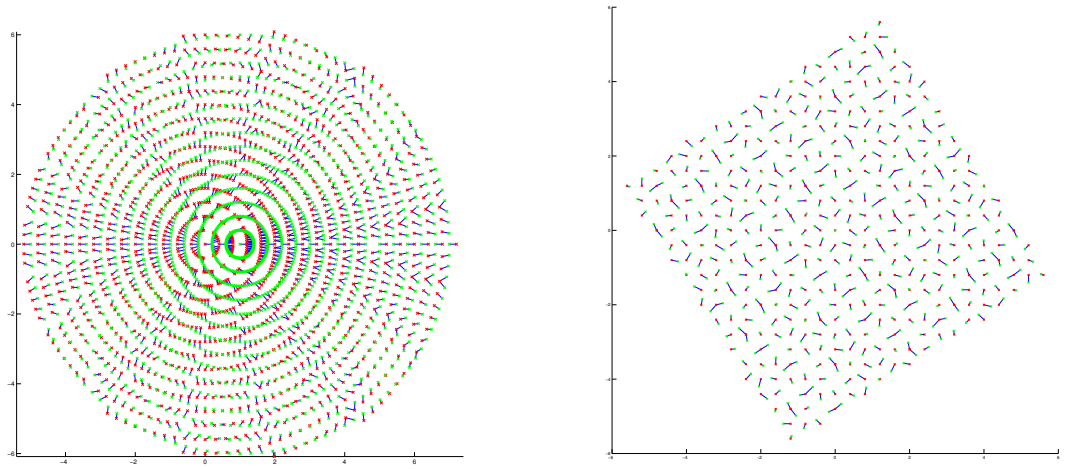
Using similar arguments, it is clear that using the log-polar grid we can also implement approximate translations. These will become more accurate as the grid becomes denser, i.e., b_0 and θ_0 become smaller.

In Figure 8.5(a), we show the nearest neighbor mapping of shifted receptive fields in one of the scales in the log-polar grid S_p. The green markers indicate the centers of the translated receptive fields. The latter are connected by the blue lines, to the red markers that represent their nearest neighbor. As a comparison, we also show the nearest neighbor mapping of rotated receptive fields in one scale in the Cartesian grid S_c in Figure 8.5(b).

8.2 Examples

The theoretical tools developed in Section 8.1 have been evaluated in a number of directions. This pertains to separable and nonseparable visual streams, and to the architecture of the video time encoders employed, including the choice of receptive fields and spiking neuron models. It also pertains to exploring the sampling of the IPT transformations, including dilations, rotations and translations. Finally, it pertains to the recovery of the encoded visual streams, including exact and approximate stimulus decoding algorithms.

We consider a space-time separable video stream of the form $u(x, y, t) = A(x, y)u(t)$, where $A(x, y)$ is an 256×256 pixel image defined on the spatial domain of $\mathbb{D} = [-8, 8] \times [-8, 8]$, and $u(t)$ is an one second long temporal signal of bandwidth 4 Hz. In the following examples, we show snapshots of the video stream u and the various transformed videos



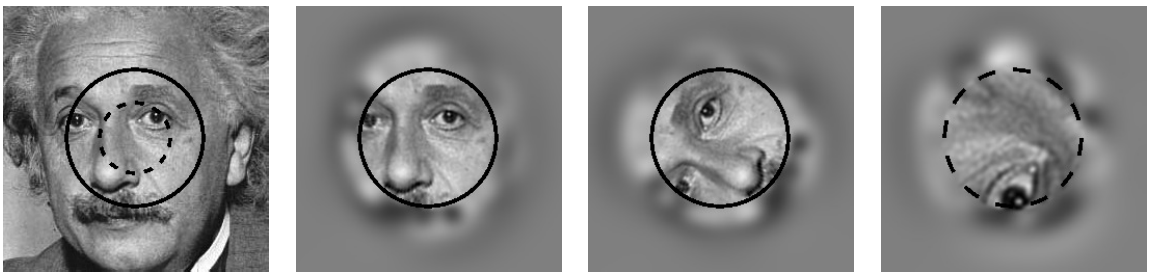
(a) (b)

Figure 8.5: **(A)** Mapping of the translated receptive field of $\alpha = 0.5$ to their approximations in the existing polar grid. The green markers indicate the translated receptive fields, the red markers indicate the approximations, and the blue line shows the mapping from green to red. **(B)** Mapping of the rotated receptive field in dilation $\alpha = 0.5$ to their approximations in the existing Cartesian grid. The green markers indicate the translated receptive fields, the red markers indicate the approximations, and the blue line shows the mapping from green to red. (Modified from [114] with permission, ©2013 Elsevier)

at the time instant $t|_{u(t)=1}$ for illustration purposes. The spatial component of the input stimulus is visualized in Figure 8.6A. The Structural Similarity (SSIM) index [190] of the shown snapshot is evaluated for each of the examples. Bilinear interpolation is used to perform transformations on the original signals to create references that the reconstructions are compared to.

In our evaluations we will also use a space-time nonseparable natural visual stream with the same characteristics as the separable video stream described above, that incidentally, exhibits a higher temporal bandwidth. The SSIM index is evaluated for the entire visual stimulus.

8.2.1 Rotations and Dilations on the Log-Polar Grid



(a) (b) (c) (d)

Figure 8.6: Rotations and dilations (zoom-in) on the log-polar grid. (a) Spatial component of the original stimulus. The reconstruction is performed only for the region inside the solid black circle. The dashed circle indicates the region that was zoomed into. (b) Recovery with a null connectivity setting switching matrix. (c) Recovery with a switching matrix connectivity setting realizing a rotation of 69 degrees counter-clockwise. (d) Recovery with a switching matrix connectivity setting realizing a rotation of 135 degree clockwise and a dilation factor of 2. The region corresponds to the one inside the dashed circle in the original stimulus in (a). The recovery was multiplied by 2 before being shown. The SSIM index for (b-d) is, respectively, $0.90, 0.93, 0.94$. (Modified from [114] with permission, ©2013 Elsevier)

We start by constructing the receptive fields on the log-polar grid and by providing the parameters of the IAF neurons defining the Video TEM. Subsequently, we describe how to achieve with the TDP block rotation and dilation transformations.

Center-surround receptive fields of RGCs and neurons in the LGN have been modeled with the Difference of Gaussians (DoG) wavelet [93, 164]. Here the mother wavelet of the receptive fields is the DoG

$$\eta(x,y) = \frac{1}{2\alpha_1^2}\exp\left(-\frac{x^2+y^2}{2\alpha_1^2}\right) - \frac{1}{2\alpha_2^2}\exp\left(-\frac{x^2+y^2}{2\alpha_2^2}\right), \tag{8.19}$$

with parameters $\alpha_1 = 0.5, \alpha_2 = 1.6\alpha_1$.

The parameters for generating the filter bank are defined on the log-polar grid (see Section 8.1.2.3):

- $\alpha_0 = 2, m \in \{-3,-2,-1,0,1\}$,

- $\theta_0 = 2\pi/L, L = 120, l \in \{0, 1, \cdots, 119\}, \omega_0 = 2\pi, N = 1,$
- $b_0 = 0.8.$

Each receptive field output was fed into an IAF neuron, with bias $b = 0.4$, threshold $\delta = 0.03$ and integration constant $\kappa = 1$. The total number of receptive fields, as well as the total number of neurons was $18,605$; a total of $245,690$ spikes were fired. We focussed on the reconstruction of the circular region $x^2 + y^2 \leq 4^2$ as indicated by the solid circle in Figure 8.6(a). All the receptive fields centered in the domain were taken into account, together with a small number of receptive fields whose centers are immediate neighbors of the circular domain.

We present several examples showing the recovery of stimuli encoded with the Video TEM described above. Two experiments are described. In the first, the encoded stimulus is space-time separable. In the second, a natural video stream is used that is non-separable.

We first demonstrate the ability of the TDP block with various connectivity settings to recover, rotate and dilate a separable visual stream. Perfect recovery was realized by the TDP block with the null setting of the switching matrix performing an identity transformation. Figure 8.6(b) shows the recovery of the visual stimulus within the framed region. Since $\theta_0 = 2\pi/120$ in the log-polar grid, any rotations of multiples of 3 degrees can be realized. In this example, a rotation of 69 degrees in the counter-clockwise direction is performed. The spike train coming from the neuron whose spatial receptive field is the element $(\alpha_0^m r_{l\theta_0}[kb_0, 0], \alpha_0^m, l\theta_0)$ of the filter bank is mapped into the reconstruction filter with parameter $(\alpha_0^m r_{(l+23)\theta_0}[kb_0, 0], \alpha_0^m, (l+23)\theta_0)$. The resulting reconstruction is shown in Figure 8.6(c).

We now present an example of a simultaneous rotation of 135 degrees clockwise and a dilation by a factor of 2. This transformation can be achieved by routing the spikes fired by the neuron whose spatial receptive field is the element $(\alpha_0^m r_{l\theta_0}[kb_0, 0], \alpha_0^m, l\theta_0)$ to the reconstruction filter $(\alpha_0^{m+1} r_{(l-45)\theta_0}[kb_0, 0], \alpha_0^{m+1}, (l-45)\theta_0)$, $m = -3, -2, -1, 0$. The spikes of the encoding neurons at scale $m = 1$ were ignored since the scale $m = 2$, at which they were to be shifted, did not exist. The result is shown in Figure 8.6(d). Note that the reconstruction in the figure had been multiplied by 2 for illustration purposes since the

dilated version is scaled by $\frac{1}{2}$ due to the unitary condition.

Second, we demonstrate rotation and dilation transformations executed by the TDM on a space-time non-separable natural visual stream, the size of which is the same as the separable video stream in the previous examples. We used the same set of receptive fields and neurons, except that the bias and threshold of each of the neurons were respectively set to $b = 1.2$ and $\delta = 0.04$. These parameters guarantee the high quality recovery of the encoded video stimuli. The ensemble of neurons fired $550,722$ spikes in the 1 second duration of the visual stream. In the video of Figure 8.7 we show the reconstructed stimulus and 2 different rotation transformations. We also show in the video of Figure 8.8 three transformations that involves dilations and rotations.

Figure 8.7: Rotations non-separable natural scenes on the log-polar grid. The original video is shown in the 1st column. In the 2nd column, the original stimulus is recovered. In the 3rd column, the recovered video is rotated counter-clockwise by 45 degree. In the 4th column, the recovered video is rotated clockwise by 171 degree.

From the results obtained in this section, it is clear that IPTs such as rotations and dilations can be performed in the spike domain with a simple switching mechanism as we have described in Section 8.1.2.3.

8.2.2 Translations and Dilations on the Cartesian Grid

We start by presenting the architecture of the Video TEM. This is followed by the TDP block connectivity settings to achieve translation and dilation transformations.

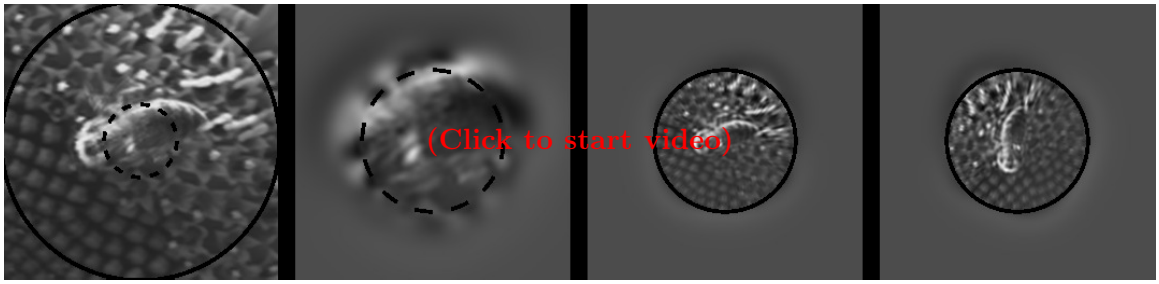


Figure 8.8: Rotations and dilations (zoom-out) of non-separable natural scenes on the log-polar grid. The original video is shown in the 1st column. In the 2nd column, the recovered video is dilated by a factor of 2. In the 3rd column, the recovered video is dilated by a factor of 0.5. In the 4th column, the recovered video is dilated by a factor of 0.5 and rotated counter-clockwise by 63 degree.

The mother wavelet that was chosen in this case was a Gabor wavelet of the form

$$\eta(x,y) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2} - \frac{y^2}{8}\right)\left(e^{i\kappa x} - e^{-\kappa^2/2}\right), \tag{8.20}$$

where real and imaginary parts are separated into two real receptive fields. The Gabor wavelet models well the linear transformations observed in the receptive fields of simple cells in V1 [83].

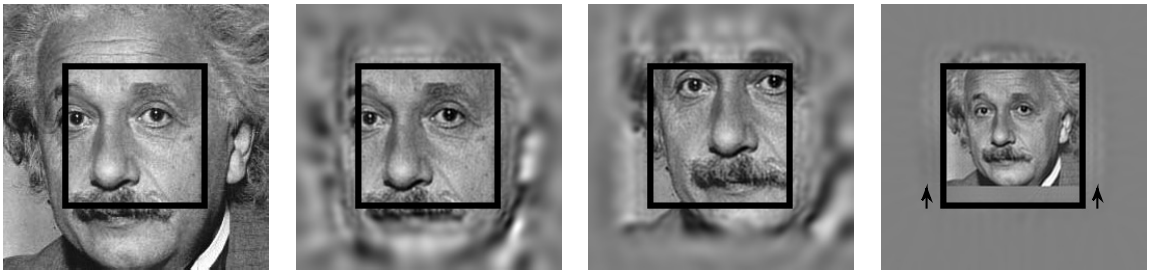
The parameters for generating the filterbank are located on the Cartesian grid (see also Section 8.1.2.4):

- $\alpha_0 = 2, m \in \{0,1,2,3\}$
- $b_0 = 2,$
- $\omega_0 = 2\pi/N, N = 8, n \in \{0,1,2,3,4,5,6,7\},$

leading to a total of 99,136 filters. Note that we also added 8 orientations for the Gabor filter, which as discussed, does not affect the translation and dilation transformations. Since the simulation can only be performed for a finite grid instead of countably infinite one, we again only consider the reconstruction inside the $[-4,4] \times [-4,4]$ region in the middle of the image, as highlighted by the rectangle in Figure 8.9(a). Receptive fields centered in this

region and in its immediate neighborhood were taken into account. All the IAF neurons had parameters $\kappa = 1, \delta = 0.02$ and $b = 0.3$.

Perfect recovery can be implemented by the null setting switching matrix performing an identity transformation in the TDP building block. Figure 8.9(b) shows the reconstruction of the visual stimulus within the framed region.



(a) (b) (c) (d)

Figure 8.9: Translations and dilations on the Cartesian grid. (a) Original stimulus. The recovery is performed only for the region inside the black frame. (b) Recovery with a switching matrix with null connectivity setting. (c) Recovery with a switching matrix connectivity setting realizing a translation by 2 units to the right and 2 units upwards, as indicated by the arrow. (d) Recovery with a switching matrix connectivity setting realizing a dilation of factor of 0.5 and a translation by 1 unit upwards, as indicated by the arrow. The recovery was multiplied by 0.5 before being shown. The SSIM index for (b-d) is, respectively, $0.95, 0.95, 0.92$. (Modified from [114] with permission, ©2013 Elsevier)

Translation by 2 units both to the right and upwards corresponds to the connectivity setting of the switching matrix that maps the receptive field representing the element $(\alpha_0^{-m}[kb_x, lb_y], \alpha_0^{-m}, n\omega_0)$ to $(\alpha_0^{-m}[(k + \alpha_0^m \cdot 1)b_x, (l + \alpha_0^m \cdot 1)b_y], \alpha_0^{-m}, n\omega_0)$. The translation result is shown in Figure 8.9(c), and the shift is indicated by the arrow.

We now demonstrate simultaneous dilation by a factor of $\frac{1}{2}$ and translation upwards by 1 unit. This was be achieved by the switching matrix setting that wires receptive fields representing the elements $(\alpha_0^{-m}[kb_x, lb_y], \alpha_0^{-m}, n\omega_0)$ to the elements $(\alpha^{-m-1}[kb_x, (\alpha_0^m + l)b_y], \alpha^{-m-1}, n\omega_0)$, for $m = 0, 1, 2$. The spikes of the neurons belonging to the dilation level $m = 3$ were ignored. The result of the simultaneous translation and dilation transformations

is shown in Figure 8.9(d).

We now show examples for transformations of non-separable natural scenes on the Cartesian grid. In the video in Figure 8.10, we show reconstruction of the original video and two different translations of the original video. In the video in Figure 8.11, 3 examples of dilations and translations of the original video are shown.

Figure 8.10: Translations of non-separable natural scenes on the Cartesian grid. The original video is shown in the 1st column. In the 2nd column, the original stimulus is recovered. In the 3rd column, the recovered video is translated downwards by 2 units and to the left by 2 units. In the 4th column, the recovered video is translated upwards by 4 units.

Figure 8.11: Translations and dilations of non-separable natural scenes on the Cartesian grid. The original video is shown in the 1st column. In the 2nd column, the recovered video is dilated by a factor of 2. In the 3rd column, the recovered video is dilated by a factor of 0.5. In the 4th column, the recovered video is dilated by a factor of 0.5 and translated to the right by 2 units.

Concluding, if the spatial receptive fields are defined on a Cartesian grid as described in Section 8.1.2.4, then admissible translation and dilation IPTs can be achieved using a simple

switching mechanism.

8.2.3 Approximate Transformations

In the previous sections we have shown that the "natural" grid for constructing the Video TEMs for rotations and dilations is the log-polar grid. We have also demonstrated that the natural grid for translations and dilations is the Cartesian grid.

In this section we explore the performance in recovery when the receptive fields of the Video TEMs are built on the log-polar grid and translation and dilation transformations are performed in the spike domain. Conversely, for receptive fields of the Video TEMs defined on the Cartesian grid, rotations and translations are considered.

Throughout, we constructed the receptive fields of the Video TEM with the DoG mother wavelet given in (8.19). The parameters of the IAF neurons are as described in Section 8.2.1.

In order to explore approximate rotation transformations, the receptive fields were placed on a Cartesian grid with the following parameters:

- $\alpha_0 = 2, m \in \{0, 1, 2, 3\}$,
- $\omega_0 = 2\pi, N = 1$,
- $b_0 = 0.8$.

We performed several clockwise rotations with successive multiples of 23°. The results are shown in Figure 8.12. Only the recovery shown in Figure 8.12(a) is exact. The other rotations in Figure 8.12(b-h) were approximate. Nevertheless, the original image can easily be identified with the desired rotations. Transformations of space-time non-separable natural scenes are shown in the video of Figure 8.13.

To illustrate the effect of the density of the Cartesian grid on arbitrary rotations, we performed three experiments with increasing grid density. By setting $b_0 = 1.0, 0.8, 0.6$ in each experiment, respectively, we rotated the video by 144° clockwise. The results are shown in Figure 8.14(b-d). The rotations were performed correctly for all three b_0's. Moreover, as we decreased b_0 from 1.0 in Figure 8.14(d), to 0.8 in Figure 8.14(c) and finally to 0.6 in Figure 8.14(b), the quality of the approximate transformation increasingly improved. We

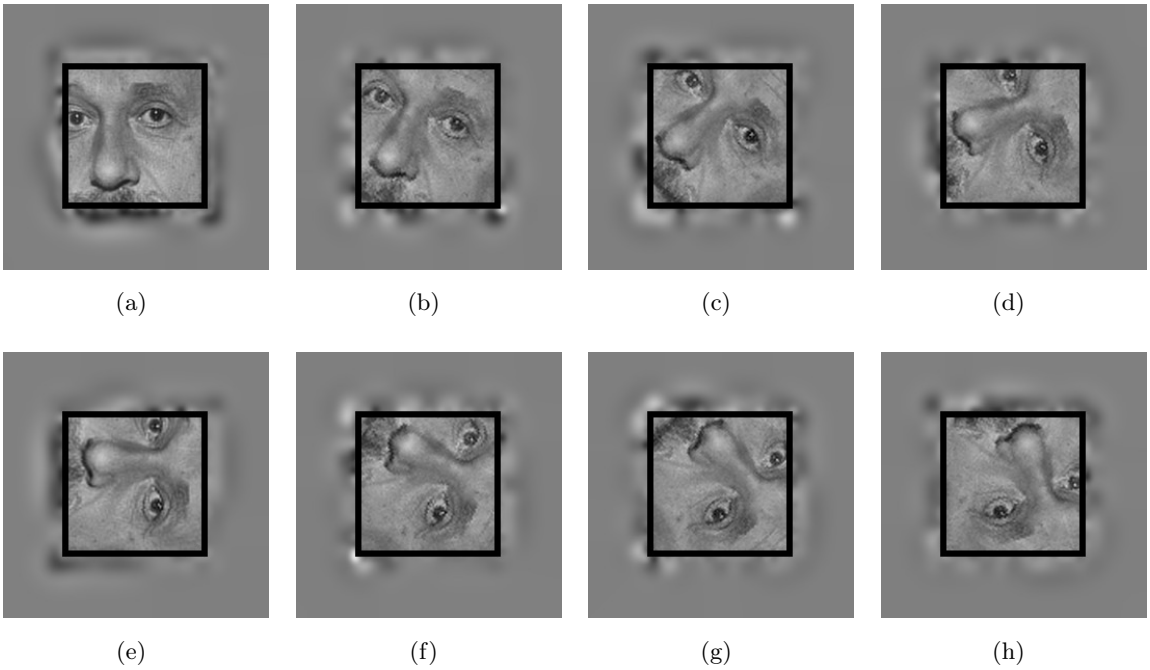


Figure 8.12: Approximate arbitrary rotations on the Cartesian grid. Recovery of a successive multiple of 23° clockwise rotations. Only the focused area of the video is being shown. (a) Recovery. (b) Rotation by 23°. (c) Rotation by 46°. (d) Rotation by 69°. (e) Rotation by 92°. (f) Rotation by 115°. (g) Rotation by 138°. (h) Rotation by 161°. The SSIM index for (a-h) is, respectively, $0.94, 0.87, 0.88, 0.87, 0.88, 0.87, 0.88, 0.86$. (Modified from [114] with permission, ©2013 Elsevier)

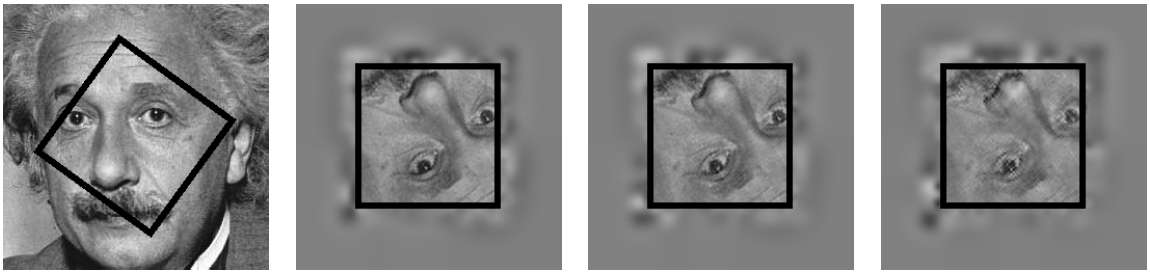
then examined the quality of stimulus recovery using the grids mentioned above. The grid density has an effect on the quality of reconstruction. Since the identity transformation can be exactly implemented, the quality degradation of the approximate rotation of stimuli parametrized by different grid densities can be further evaluated. The evaluation is based on comparing for each stimulus the approximate rotation with the identity transform on the same grid. For $b_0 = 0.6$, the quality of recovery of the approximate rotation only differs from the quality of the identity transform by 0.01 SSIM. For $b_0 = 0.8$, the quality of recovery of the rotation is lower than that of the identity transformation by 0.07 SSIM, while for $b_0 = 1.0$, the difference is 0.13 SSIM. Therefore, as the grid becomes denser, the approximate transformation converges to a "faithful" one.

Figure 8.13: Approximate Rotations of non-separable natural scenes on the Cartesian grid. The original video is shown in the 1st column. In the 2nd column, the recovered video is rotated clockwise by 46 degrees. In the 3rd column, the recovered video is rotated clockwise by 92 degrees. In the 4th column, the recovered video is rotated clockwise by 138 degrees.

Remark 21. *In practice approximate rotations of arbitrary degree can be performed using the nearest neighbor mapping described in Section 8.1.2.5. Note, however, that rotations of very similar degree, can potentially lead to the same mapping and therefore become indiscriminable, a phenomenon that has been reported in the psychophysics literature [195]. The rotation angle resolution depends on the resolution of the filterbank, i.e., the parameters* b_x, b_y *and* α_0.

In order to evaluate approximate translations on the log-polar grid we used the same encoding architecture as in the examples in Section 8.2.1. To perform a translation, the receptive fields are mapped to the nearest existing receptive field after the translation. In Figure 8.15(b-d) we show, respectively, results obtained by translation to the right by $0.5, 1$ and 1.5 units. The same translations on the non-separable natural scenes was also performed and is approximate translations are shown in the video in Figure 8.16.

We now investigate the effect of the density of the grid on the reconstruction. There are two ways to increase the density of the grid. One is to use a smaller b_0, the other is to decrease θ_0; we show their effects in Figure 8.17. Again, we can observe that the denser the grid is, the better the reconstruction quality of image (or natural scenes) translations. For $b_0 = 0.6, \theta_0 = 2\pi/180$, the recovery is less noisy. As before, since the reconstruction quality may also depend on the grid density itself, we tested the identity transformation on the same grids. The difference between the SSIM index of the identity transform and



(a) (b) (c) (d)

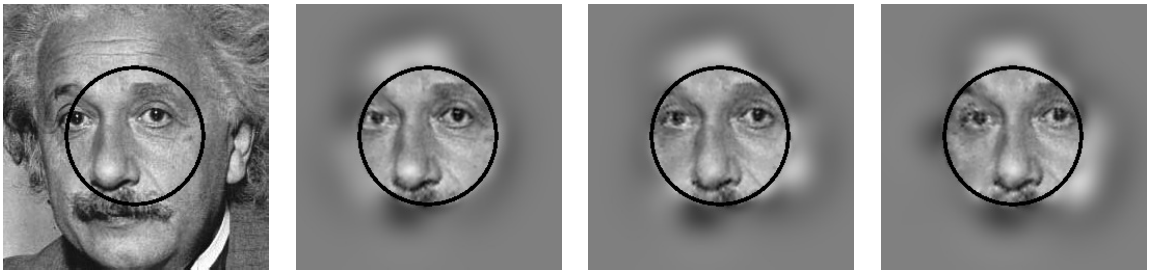
Figure 8.14: Effect of grid density on rotations with nearest-neighbor mapping on the Cartesian grid. (a) Original stimulus. (b-d) Recoveries of rotation by 144° with grid parameter (b) $b_0 = 0.6$, (c) $b_0 = 0.8$, (d) $b_0 = 1.0$. As the grid becomes denser (smaller b_0) the quality of the approximate rotation improves. The SSIM index for (b-d) is, respectively, 0.94, 0.88, 0.76. For comparison, the SSIM indexes of the recovery are $0.95, 0.94, 0.89$ for the grids in (b-d), respectively. (Modified from [114] with permission, ©2013 Elsevier)

the approximate translations decreases as the grid becomes denser. However, it should be noted that the recovery quality of the approximate translation depends not only on the grid density, but also on the value of the translation performed, since the grid is denser in the center and coarser away from the center. These observations are consistent with several experimental studies [91].

It can be seen that, translations on the polar grid are not performing nearly as well for coarser grids, especially when compared with the reconstruction quality of rotations on a Cartesian grid. Such an effect is expected, as the polar grid is not uniformly tiling the space.

8.3 Discussion and Summary

In this chapter we presented a general model for the realization of identity-preserving transformations in the spike domain. Our model architecture consists of a spike domain switching circuit (Time Domain Processing) that channels the spikes from the sensory neurons (Time Encoding Machine) into the higher brain areas (processed with the Time Decoding Machine). Surprisingly, a simple rewiring strategy can perform a class of identity-preserving transformations such as rotations, scalings and translations, thereby giving rise to a family



(a) (b) (c) (d)

Figure 8.15: Approximate translations on a log-polar grid. (a) Original stimulus. (b-d) Recovery of translations by 0.5 units in (b), by 1 unit in (c) , and by 1.5 units in (d). Smaller translations exhibit improved quality of recovery because of the non-uniform tiling of the log-polar grid. The SSIM index for (b-d) is, respectively, $0.81, 0.80, 0.79$. (Modified from [114] with permission, ©2013 Elsevier)

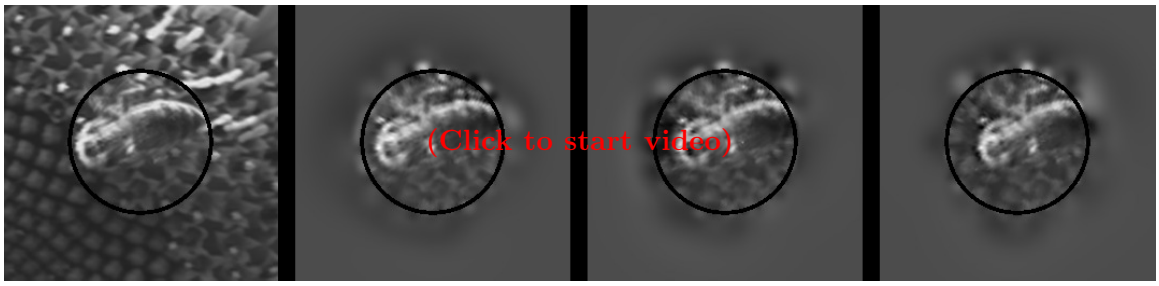
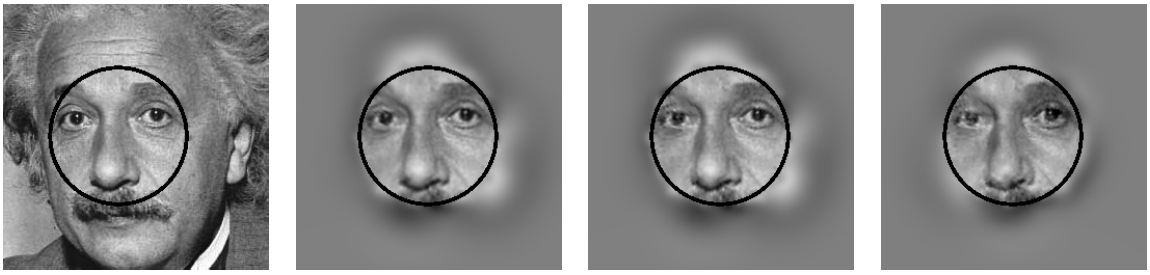


Figure 8.16: Approximate Rotations of non-separable natural scenes on the Cartesian grid. The original video is shown in the 1st column. In the 2nd column, the recovered video is translated to the right by 0.5 unit. In the 3rd column, the recovered video is translated to the right by 1 unit. In the 4th column, the recovered video is translated to the right by 1.5 unit.

of invariant transformations in the spike domain. We demonstrated that this class of transformations can easily be realized with a neural circuit architecture using the same basic stimulus decoding algorithm. What changes in the architecture are only the connectivity settings of the switching matrix (i.e., the input/output "wiring") of the TDP building block. Each connectivity setting corresponds to a particular IPT.

The t-transform formalism allows us to interpret neural spiking in the language of inner



(a) (b) (c) (d)

Figure 8.17: Effect of polar grid density on translations. (a) Original stimulus. (b-d) Recovery of translations by 1 unit, the grid densities are: (b) $(b_0 = 0.6, \theta_0 = 2\pi/180)$, (c) $(b_0 = 0.6, \theta_0 = 2\pi/120)$, (d) $(b_0 = 1.0, \theta_0 = 2\pi/120)$. As the grid becomes denser (smaller θ_0) the quality of the approximate translation improves. The SSIM index for (b-d) is, respectively, $0.85, 0.83, 0.75$. For comparison, the SSIM indexes are $0.90, 0.90, 0.89$ for the grids in (b-d), respectively. Compare also with Figure 8.15(c) where the grid density is $(b_0 = 0.8, \theta_0 = 2\pi/120)$. (Modified from [114] with permission, ©2013 Elsevier)

product operations. This formalism also enables us to consider an architecture that has provably high performance characteristics. We note that the implementation of IPTs with a switching mechanism depends only on the structure of the encoder (in our case filterbank models of receptive fields and IAF neurons for the spiking mechanisms) and not on the precise timing of spike responses. As a result, IPTs are also realizable using the same switching mechanism for any model with the same encoding structure.

When dealing with exact IPTs, a key property we have implicitly utilized is that the receptive field outputs are invariant to transformations performed on the grid. However, a common pitfall of wavelet representation of signals is that the representation is highly dependent on the relative alignment of the input signal with the grid. For example, rotation invariance is guaranteed only for rotation that are multiples of the grid size. Therefore, the invariance is discretized as well.

Shiftable and steerable filters have been introduced as an alternative approach to these problems [58, 173]. Shiftable and steerable filters provide efficient implementations of arbitrarily translated or oriented filters from a linear combination of a bank of basis filters.

These linear transformations can be viewed as coordinate transformations in the Hilbert space; a set of coordinates correspond to a collection of filters. The coordinate transformation is achieved via a linear operation (matrix multiplication). The matrix is typically obtained by solving systems of equations. The steerable and shiftable filters are carefully designed so that the matrix can be efficiently computed analytically or expressed in closed form.

Can these desirable characteristics be translated to our setting? A critical assumption for coordinate transformations, regardless of their efficiency, is that the sampling is linear. However, temporal sampling with neural circuits is highly nonlinear and signal dependent. While in our current setting linear interpolation is not readily possible, the efficiency gain of the TDM realization is even more noteworthy.

The switching matrix of the TDP block provides a simple, yet powerful realization of spike domain processing. Although the connectivity settings of the switching matrices we presented are fixed and each corresponds to a single transformation, the rewiring capability can dramatically increase the representational and processing power of the neural assemblies.

A question that naturally arises is how can such a rewiring mechanism be implemented with neural substrates and how the visual system decides which IPT to perform. A model for dynamic regulation of the connectivity between two neural layers was proposed by [141], where a set of control neurons was used to dynamically modify the synaptic strengths of the inter-layer connections. Testing this hypothesis, however, would require tracking the functional switching of a large population of neurons at a very fast timescale. Given the current and near term experimental capabilities, a testable prediction about the switching connectivity matrix performed on large populations, while desirable, is out of reach.

The IPTs implemented with the switching mechanism can play an important role in view-dependent invariant recognition. In the class of invariant recognition models where transformations of the incoming visual signal are matched with an existing stored version of the image, object representations are stored as originally viewed [16]. Recognition is achieved by transforming the input to match the view specification of the stored representation. These transformations can be achieved by interpolation [152], mental transformation [181] or alignment [184]. In our setting, these transforms can be readily achieved by multiple

parallel readouts corresponding to multiple transforms executed in parallel.

A way of determining the transformation between the input and the stored object has been proposed by [6]. Called the map seeking circuit (MSC), this algorithm identifies a discretely parametrized linear transformation (based on rotations, translations, dilations, etc.) that minimizes an appropriate cost functional. To do the MSC algorithm iteratively, three classes of operations have to be specified: (i) a set of linear transformations, (ii) evaluation of the similarity between a stored version of an image and the transformed input image, and (iii) arithmetic computations such as addition, and appropriate updates of the coefficients of the linear transformations. Although not specifically focussing on the implementation of the MSC algorithm, our formalism provides a methodology for the parallel construction of the set of linear transformations as required by one of the key operations of the algorithm, in the context of our spiking architecture. Efficient spike domain algorithms that evaluate the similarity between transformations of the incoming stimulus and stored in memory patterns will be pursued in future work.

The set of all possible IPTs and the corresponding invariant representations that can be realized with our architecture originate from the spatial structure of the neuron receptive fields. The latter form an overcomplete spatial filterbank and also exhibit specific symmetry properties that enable the implementation of IPTs with a simple switching mechanism. By employing a similar group structure for the temporal receptive fields, one can devise a space-time overcomplete filterbank at the receptive field level, similar to the ones used in space-time wavelets [5]. The use of space-time non-separable receptive fields [98, 99] is also possible provided the group structure is kept. Such an organization of receptive fields can facilitate a richer set of IPTs that can be realized with the same switching matrix architecture, including space-time transformations. Consequently, new forms of invariant transformations can arise such as velocity invariance. The latter suggests investigating attention models for tracking.

Our focus on invertible transformations and their implementation (and the characterization of implementable IPTs) provides a solid foundation for the theoretical capabilities of our model. In addition, some classes of IPTs, such as the ones creating the mirroring effect, are also realizable under this architecture. The switching mechanism can also be used for non-invertible transformations. For example by combining the operations of scaling

and translation one can zoom into a particular region of the incoming video stream, and thereby select a particular spatial region to be propagated to the next layer. This suggests a methodology for the implementation of attention-selective mechanisms. Such properties also address the correspondence problem of identifying equivalent stimuli while constantly changing visual fixations. A further example of non-invertible transformations arises when the encoder is noisy. In the latter case, the mathematical formalism for signal recovery is based on regularization [102, 112] and can be directly adapted to our setting. However, other IPTs that frequently arise in visual recognition tasks that facilitate invariance to, for example, occlusion, clutter and illumination will require additional mechanisms.

In our architecture IPTs are realized by processing spikes, the natural language of the brain. Spike processing in the TDP block is based on a key symmetry assumption on the receptive fields. It is also based on the assumption that the spiking mechanisms of the encoding neurons are identical. Consequently, IPTs can be implemented without modifying the decoding block. To overcome the latter limitation, employing further spike processing in addition to connectivity changes may be necessary, and will be investigated elsewhere.

Chapter 9

Conclusions and Future Work

9.1 Conclusions

In this dissertation, we have comprehensively investigated a class of massively parallel spiking neural circuit architectures in modeling the representation of natural visual stimuli by spike times in early visual systems. Under this architecture, neurons are modeled as dendritic stimulus processors cascaded with spike generators. The results in the thesis suggest that the class of neural circuit architectures accommodate rigorous study of visual representation on a massive scale, matching that of early visual systems.

We built on the key insight that spikes generated by neurons can be interpreted as projections of the stimulus on stimulus-dependent sampling functions [151] and showed that visual stimuli encoded by the massively parallel neural circuits with biophysically realistic neuron models can be formulated similarly by introducing a PRC manifold. In addition, we demonstrated that a visual stimulus encoded by a massively parallel neural circuit can be perfectly reconstructed and that conditions for it can be specified. Extension to include ON-OFF DSPs and feedbacks largely eliminated the need for bias current in BSGs, reducing the power budget of these circuit architectures when implemented in silicon.

We also investigated the representation of visual stimuli with DSPs of different forms, such as the representation of stereoscopic color visual stimuli with color-component DSPs as well as the representation of visual stimuli by an ensemble of V1 complex cells with Volterra

DSPs. The result related to the encoding of stereoscopic color visual stimuli suggests that color or binocular information can be readily encoded by the massively parallel neural circuits and that it is possible to demix information in different channels from an unlabeled set of spike trains. Our result on neural circuits with Volterra DSPs suggests that certain nonlinear visual processing in early vision does not affect the faithful representation of stimuli by an ensemble of neurons. Rather, the representation is different, allowing the extraction of diverse features while keeping the total visual information intact.

By the duality between decoding and functional identification, explicit algorithms were proposed for functionally identifying massively parallel neural circuit of various types, including those encoding color/stereoscopic visual stimuli and those modeling visual encoding by an ensemble of complex cells.

A key methodology in relating intrinsic neural noise sources and neural representation was advanced in this dissertation. In contrast to existing methods such as those using point process to directly construct uncertainty in the spike generation mechanism, our method relied on first building a dynamical system with a biophysical spiking mechanism modeled as ordinary differential equations and then introducing naturally intrinsic noise sources with stochastic differential equations. This allowed us to formulate the encoding under noisy conditions as generalized sampling with noisy measurements and to systematically investigate the effect of noise on both encoding/decoding and functional identification.

By using group theoretic methods, we introduced the identity preserving transformation, a type of visual stimuli processing, in the spike domain. The extensive demonstration of exact and approximate IPTs using two neural encoding architectures showed the effectiveness and simplicity of the “switching matrix" in visual stimulus processing. The ability to employ such processing with the nonlinearity in the spiking mechanism was surprising.

Finally, constructing an “oscilloscope" using decoding not only enabled the reconstruction of stimuli in order to probe the information content encoded or processed by a neural circuit, but also provided an intuitive method of evaluating the quality of functional identification of a massively parallel neural circuit.

9.2 Directions of Future Research

The work presented in this dissertation opens up several avenues for future research on spike-based representation and processing of visual stimuli with massively parallel neural circuit architectures.

As discussed in Chapter 4, one question arising in the representation of stereoscopic color video in the spike domain is how to more efficiently represent different color/eye channels by observing the commonality of these channels in a single visual scene. For example, color information is often redundant, and a visual scene may be constructed from a fine representation of gray-scale visual scene plus a coarser representation of its color. The color information can be "inpainted" from the full gray-scale information and the partial color information [157]. This seems to be the strategy that the retina is taking, as more rods are present than cones in the area outside fovea. In addition, this will allow the retina to reduce the information rate sent to the cortex.

The same question can be applied to stereoscopic vision, in which both eyes see the same visual scene. Moreover, it has been suggested that the response from complex cells can provide a reasonable estimation of binocular disparity [155, 156]. It would be interesting to study how depth information as well as the light intensity can both be recovered from the spikes generated by massively parallel neural circuits in encoding.

The study of the visual representation by an ensemble of complex cells points out several interesting questions to be investigated. First, can the nonlinear interaction be applied to the color-component DSPs such that the abovementioned encoding strategy can be more easily realizable? The second question that naturally arises is why the cortex employs two or even more representations, *e.g.*, a representation with simple cells and a representation of complex cells. What is the advantage of doing so, and what are the individual roles of simple and complex cells? Third, as we have shown that reconstruction from the outputs of either simple cells or complex cells is possible, it is advisable to think that utilizing both types of cells in V1 can lead to a faithful representation of the scene, and the visual scene can be reconstruction from a different but closely related optimization problem. The question, however, is what mixture is sufficient to guarantee faithful representation.

We investigated visual processing in the spike domain in Chapter 8. As we have worked out quite a few types of representations, it becomes interesting to determine what other processing can be achieved in the spike domain. In Chapter 6, we introduced a class of Volterra DSPs whose inputs are spikes generated by the BSGs. The functionality of these Volterra DSPs can be applied not only to feedbacks as described in Chapter 6, but also to feedforward processing, *i.e.*, they become the input DSPs for the neurons in the next layer that receive spiking inputs [104]. These Volterra DSPs can lead to interesting nonlinear spike processing directly in the spike domain [126].

The main focus of this dissertation has been on the algorithmic level of information representation and processing. It is natural to ask how these algorithms can be implemented by neural components. The retina can obviously provide insights into this question given the amount of anatomical knowledge generated in the past decade [14]. It is, however, also interesting to investigate the visual systems in invertebrates, particularly, in fruit flies.

The fruit fly provides several advantages in studying visual representation and processing. First, the size of the fruit fly's early visual system is much smaller than that of the vertebrate retina. Second, the full connectome of the fruit fly's visual system may become the first fully reconstructed visual system [163, 180]. Third, extensive genetic tools available for fruit flies may lead to faster progress in experiments to verify the functions of circuits [25, 139].

The fruit fly has compound eyes that make it possible to devise parallel programming models [115] by the observation of canonical neural circuit abstractions in several visual neuropils [113]. This may provide guidance in the study of parallel visual processing in general. By building a circuit-level simulation [66, 116, 117], this opens an avenue to link the algorithmic level and implementation level together. A parallel programming model on the implementation-level can help to advance algorithmic-level understanding of the visual system, while the algorithmic-level study may guide the development of implementations in biological settings.

Bibliography

[1] Edward H. Adelson and James R. Bergen. Spatiotemporal energy models for the perception of motion. *Journal of the Optical Society of America. A: Optics and image science*, 2(2):284–299, 1985.

[2] Hagai Agmon-Snir, Catherine E. Carr, and John Rinzel. The role of dendrites in auditory coincidence detection. *Nature*, 393:268–272, 1998. doi: 10.1038/30505.

[3] Blaise Agüera y Arcas, Adrienne L. Fairhall, and William Bialek. Ccomputation in a signal neuron: Hodgkin and Huxley revisited. *Neural Computation*, 15:1715–1749, 2003. doi: 10.1162/08997660360675017.

[4] Charles H. Anderson and David C. Van Essen. Shifter circuits: A computational strategy for dynamic aspects of visual processing. *Proceedings of the National Academy of Sciences*, 84:6297–6301, 1987.

[5] Jean-Pierre Antoine, Romain Murenzi, Pierre Vandergheynst, and Syed Twareque Ali. *Two-Dimensional Wavelets and their Relatives*. Cambridge University Press, 2004.

[6] David W. Arathorn. *Map-Seeking Circuits in Visual Cognition*. Stanford University Press, Stanford, CA, 2002.

[7] Stephen A. Baccus and Markus Meister. Fast and slow contrast adaptation in retinal circuitry. *Neuron*, 36:909–919, 2002. doi: 10.1016/S0896-6273(02)01050-4.

[8] Tom Baden, Philipp Berens, Matthias Bethge, and Thomas Euler. Spikes in mammalian bipolar cells support temporal layering of the inner retina. *Current Biology*, 23:48–52, 2013. doi: 10.1016/j.cub.2012.11.006.

[9] Ethan A. Benardete and Ehud Kaplan. The receptive field of the primate P retinal ganglion cell, I: Linear dynamics. *Visual Neuroscience*, 14:169–185, 1998.

[10] Alain Berlinet and Christine Thomas-Agnan. *Reproducing Kernel Hilbert Spaces in Probability and Statistics*. Kluwer Academic Publishers, 2004.

[11] Michael J. Berry, David K. Warland, and Markus Meister. The structure and precision of retinal spike trains. *Proceedings of the National Academic of Sciences of the United States of America*, 94:5411–5416, 1997.

[12] Anatoly Yu. Bezhaev and Vladimir A. Vasilenko. *Variational Theory of Splines*. Kluwer Academic / Plenum Publishers, New York, 2001.

[13] Colin Blakemore, Roger H. S. Carpenter, and Mark A. Georgeson. Lateral inhibition between orientation detectors in the human visual system. *Nature*, 228:37–39, 1970. doi: 10.1038/228037a0.

[14] Alexander Borst and Thomas Euler. Seeing things in motion: Models, circuits, and mechanisms. *Neuron*, 71:974–994, 2011.

[15] Alexander Borst, Juergen Haag, and Dierk F. Reiff. Fly motion vision. *Annual Review of Neuroscience*, 33:49–70, 2010. doi: 10.1146/annurev-neuro-060909-153155.

[16] Heinrich H. Bülthoff and Shimon Edelman. Psychophysical support for a two-dimensional view interpolation theory of object recognition. *Proceedings of the National Academy of Sciences*, 89:60–64, 1992.

[17] Daniel A. Butts, Chong Weng, Jianzhong Jin, Chun-I Yeh, Nicholas A. Lesica, Jose-Manuel Alonso, and Garrett B. Stanley. Temporal precision in the neural code and the timescales of natural vision. *Nature*, 449:92–95, 2007. doi: 10.1038/nature06105.

[18] William H. Calvin and Charles F. Stevens. Synaptic noise and other sources of randomness in motoneuron interspike intervals. *Journal of Neurophysiology*, 31(4):574–587, 1968.

[19] Carmen C. Canavier. Phase responses curve. *Scholarpedia*, 1(12):1332, 2006. doi: 10.4249/scholarpedia.1332.

[20] Emmanuel J. Candès, Yonina C. Eldar, Thomas Strohmer, and Vladislav Voroninski. PhaseLift: Exact and stable signal recovery from magnitude measurements via convex programming. *Communications in Pure and Applied Mathematics*, 66(8):1241–1274, 2011. doi: 10.1002/cpa.21432.

[21] Andrea Caponnetto, Charles A. Micchelli, Massimiliano Pontil, and Yiming Ying. Universal multi-task kernels. *Journal of Machine Learning Research*, 9:1615–1646, 2008.

[22] Matteo Carandini, jonathan B. Demb, Valerio Mante, David J. Tolhurst, Yang Dan, Bruno A. Olshausen, Jack L. Gallant, and Nicole C. Rust. Do we know what the early visual system does? *Journal of Neuroscience*, 25(46):10577–10597, 2005. doi: 10.1523/JNEUROSCI.3726-05.2005.

[23] Claudio Carmeli, Ernesto De Vito, and Alessandro Toigo. Vector valued reproducing kernel Hilbert spaces of integrable functions and mercer theorem. *Analysis and Applications*, 4(4):377–408, 2006. doi: 10.1142/S0219530506000838.

[24] Divya Chander and E. J. Chichilnisky. Adaptation to temporal contrast in primate and salamander retina. *Journal of Neuroscience*, 21(24):9904–9916, 2001.

[25] Ann-Shyn Chiang, Chih-Yung Lin, Chao-Chun Chuang, Hsiu-Ming Chang, Chang-Huain Hsieh, Chang-Wei Yeh, Chi-Tin Shih, Jian-Jheng Wu, Guo-Tzau Wang, Yung-Chang Chen, Cheng-Chi Wu, Guan-Yu Chen, Yu-Tai Ching, Ping-Chang Lee, Chih-Yang Lin, Hui-Hao Lin, Chia-Chou Wu, Hao-Wei Hsu, Yun-Ann Huang, Jing-Yi Chen, Hsin-Jung Chiang, Chun-Fang Lu, Ru-Fen Ni, Chao-Yuan Yeh, and Jenn-Kang Hwang. Three-dimensional reconstruction of brain-wide wiring networks in drosophila at single-cell resolution. *Current Biology*, 21(1):1–11, January 2011. ISSN 1879-0445. doi: 10.1016/j.cub.2010.11.056.

[26] E. J. Chichilnisky. A simple white noise analysis of neuronal light responses. *Network: Computation in Neural Systems*, 12:199–213, 2001.

[27] Shyamprasad Chikkerur, Vijay Sundaram, Martin Reisslein, and Lina J. Karam. Objective video quality assessment methods, a classification, review, and performance comparison. *IEEE Transactions on Broadcasting*, 57(2):165–182, 2011. doi: 10.1109/TBC.2011.2104671.

[28] Leon Chua, Valery Sbitnev, and Hyongsuk Kim. Hodgkin-huxley axon is made of memristors. *International Journal of Bifurcation and Chaos*, 22(3):1230011, 2012. doi: 10.1142/S021812741230011X.

[29] A. Cichochi and R. Unbehauen. *Neural Networks for Optimization and Signal Processing*. John Wiley & Sons, 1993.

[30] Damon A. Clark, Limor Burszlyn, Mark A. Horowitz, Mark J. Schnitzer, and Thomas R. Clandinin. Defining the computational structure of the motion detector in *Drosophila*. *Neuron*, 70:1165–1177, 2011. doi: 10.1016/j.neuron.2011.05.023.

[31] J. A. Connor and C. F. Stevens. Prediction of Repetitive Firing Behaviour from Voltage Clamp Data on an Isolated Neurone Soma. *Journal of Physiology*, 213:31–53, 1971.

[32] Lisa J. Croner and Ehud Kaplan. Receptive fields of P and M ganglion cells across the primate retina. *Vision Research*, 35(1):7–24, 1995. doi: 10.1016/0042-6989(94)E0066-T.

[33] Rava Azeredo da Silveira and Botond Roska. Cell types, circuits, computation. *Current Opinion in Neurobiology*, 21:664–671, 2011. doi: 10.1016/j.conb.2011.05.007.

[34] Dennis M. Dacey. Parallel pathways for spectral coding in primate retina. *Annual Review Neuroscience*, 23:743–775, 2000. doi: 10.1146/annurev.neuro.23.1.743.

[35] Ingrid Daubechies. *Ten Lectures on Wavelets*. Society for Industrial and Applied Mathematics, 1992.

[36] Peter Dayan and L.F. Abbott. *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*. MIT Press, Cambridge, MA, 2001.

[37] Rob R. de Ruyter van Steveninck, Geoffrey Lewen, Steven P. Strong, Roland Koberle, and William Bialek. Reproducibility and variability in neural spike trains. *Science*, 275:1805–1808, 1997. doi: 10.1126/science.275.5307.1805.

[38] A. Destexhe, M. Rudolph, J.-M. Fellous, and T. J. Sejnowski. Fluctuating synaptic conductances recreate in vivo-like activity in neocortical neurons. *Neuroscience*, 107 (1):13–24, 2001. doi: 10.1016/S0306-4522(01)00344-X.

[39] A. Destexhe, M. Rudolph, and D. Paré. The high-conductance state of neocortical neurons in *vivo*. *Nature Reviews Neuroscience*, 4:739–751, 2003. doi: 10.1038/nrn1198.

[40] Alain Destexhe and Michelle Rudolph-Lilith. *Neuronal Noise*, volume 8 of *Springer Series in Computational Neuroscience*. Springer, New York, NY, 2012. doi: 10.1007/978-0-387-79020-6.

[41] James J. DiCarlo and David D. Cox. Untangling invariant object recognition. *Trends in Cognitive Sciences*, 11(8):333–341, 2007. doi: 10.1016/j.tics.2007.06.010.

[42] Peter C. Dodwell. The Lie transformation group model of visual perception. *Attention, Perception, & Psychophysics*, 34(1):1–16, 1983. doi: 10.3758/BF03205890.

[43] Steffen E. Eikenberry and Vasilis Z. Marmarelis. A nonlinear autoregressive volterra model of the hodgkin-huxley equations. *Journal of Computational Neuroscience*, 34 (1):163–183, 2012. doi: 10.1007/s10827-012-0412-x.

[44] Chris Eliasmith and Charles H. Anderson. *Neural engineering: Computation, representation, and dynamics in neurobiological systems*. MIT Press, Cambridge, MA, 2003.

[45] Christina Enroth-Cugell and J. G. Robson. The contrast sensitivity of retinal ganglion cells of the cat. *Journal of Physiology*, 187:517–552, 1966.

[46] B. Ermentrout. Type I membranes, phase resetting curves, and synchrony. *Neural Computation*, 8(5):979–1001, Jul 1996.

[47] G. B. Ermentrout and N. Kopell. Oscillator death in systems of coupled neural oscillators. *SIAM Journal of Applied Mathematics*, 50(1):125–146, 1990. doi: 10.1137/0150009.

[48] G. Bard Ermentrout and David H. Terman. *Mathematical Foundations of Neuroscience*. Springer, 2010.

[49] G. Bard Ermentrout, Galán, and Nathaniel N. Urban. Relating neural dynamics to neural coding. *Physical Review Letters*, 99(24):248103, 2007. doi: 10.1103/PhysRevLett.99.248103.

[50] A. Aldo Faisal and Simon B. Laughlin. Stochastic simulations on the reliability of action potential propagation in thin axons. *PLoS Computational Biology*, 3(5):e79, 2007. doi: 10.1371/journal.pcbi.0030079.

[51] A. Aldo Faisal and Ali Neishabouri. Axonal noise as a source of synaptic variability. *PLOS Computational Biology*, 10(5):e1003615, 2014. doi: 10.1371/journal.pcbi.1003615.

[52] A. Aldo Faisal, Luc P. J. Selen, and Daniel M. Wolpert. Noise in the nervous system. *Natural Review Neuroscience*, 9:292–303, 2008. doi: 10.1038/nrn2258.

[53] M. Fazel, H. Hindi, and S. Boyd. Rank minimization and applications in system theory. In *Proceedings American Control Conference*, pages 3273–3278, June 2004.

[54] J.-M. Fellous, M. Rudolph, A. Destexhe, and T. J. Sejnowski. Synaptic background noise controls the input/output characteristics of single cells in an *in Vitro* model of *in Vivo* activity. *Neuroscience*, 122:811–829, 2003. doi: 10.1016/j.neuroscience.2003.08.027.

[55] G. D. Field and E. J. Chichilnisky. Information processing in the primate retina: Circuitry and coding. *Annual Review of Neuroscience*, 30:1–30, 2007. doi: 10.1146/annurev.neuro.30.051606.094252.

[56] Jeffrey D. Fitzgerald, Ryan J. Rowekamp, Lawrence C. Sincich, and Tatyana O. Sharpee. Second order dimensionality reduction using minimum and maximum mutual information models. *PLoS Computational Biology*, 7(10):e1002249, 2011. doi: 10.1371/journal.pcbi.1002249.

[57] Ralph D. Freeman and Izumi Ohzawa. On the neurophysiological organization of binocular vision. *Vision Research*, 30(11):1661–1676, 1990. doi: 10.1016/0042-6989(90)90151-A.

[58] William T. Freeman and Edward H. Adelson. The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(9):891–906, September 1991.

[59] Uwe Friederich, Stephen A. Billings, Mikko Juusola, and Daniel Coca. We now know what fly photoreceptors compute. In *Abstracts from the Twenty Second Annual Computational Neuroscience Meeting: CNS*2013*, Paris, France, Jul 2013. doi: 10.1186/1471-2202-14-S1-O5.

[60] Fabrizio Gabbiani and Steven J. Cox. *Mathematics For Neuroscientists*. Academic Press, 2010.

[61] Jeffrey L. Gauthier, Greg D. Field, Alexander Sher, Martin Greschner, Jonathon Shlens, Alan M. Litke, and E. J. Chichilnisky. Receptive fields in primate retina are coordinated to sample visual space more uniformly. *PLoS Biology*, 7(4):e1000063, 2009. doi: 10.1371/journal.pbio.1000063.

[62] Karl R. Gegenfurtner. Cortical mechanisms of colour vision. *Nature Reviews Neuroscience*, 4, 2003. doi: 10.1038/nrn1138.

[63] Karl R. Gegenfurtner and Daniel C. Kiper. Color vision. *Annual Review Neuroscience*, 26:181–206, 2003. doi: 10.1146/annurev.neuro.26.041002.131116.

[64] Karl R Gegenfurtner and J. Rieger. Sensory and cognitive contributions of color to the recognition of natural scenes. *Current Biology*, 10(13), 2000.

[65] Wulfram Gerstner and Werner M. Kistler. *Spiking Neuron Models - Single Neurons, Populations, Plasticity*. Cambridge University Press, 2002.

[66] Lev E. Givon and Aurel A. Lazar. Neurokernel: An Open Scalable Software Framework for Emulation and Validation of Drosophila Brain Models on Multiple GPUs. *Neurokernel RFC #1*, pages 1–23, 2014.

[67] Joshua H. Goldwyn and Eric Shea-Brown. The what and where of adding channel noise to the hodgkin-huxley equations. *PLoS Computational Biology*, 7(11):e1002247, 2011. doi: 10.1371/journal.pcbi.1002247.

[68] Joshua H. Goldwyn, Nikita S. Imennov, Michael Famulare, and Eric Shea-Brown. Stochastic differential equation models for ion channel noise in hodgkin-huxley neurons. *Physics Review E*, 83(4):041908, 2011. doi: 10.1103/PhysRevE.83.041908.

[69] Tim Gollisch and Markus Meister. Rapid neural coding in the retina with relative spike latencies. *Science*, 319:1108–1111, 2008. doi: 10.1126/science.1149639.

[70] Tim Gollisch and Markus Meister. Eye smarter than scientists believed, neural computations in circuits of the retina. *Neuron*, 65:150–164, 2010. doi: 10.1016/j.neuron.2009.12.009.

[71] W. Govaerts and B. Sautois. Computation of the phase response curve: A direct numerical approach. *Neural Computation*, 18:817–847, 2006. doi: 10.1162/neco.2006.18.4.817.

[72] Daniel J. Graham, Damon M. Chandler, and David J. Field. Can the theory of "whitening" explain the center-surround properties of retinal ganglion cell receptive fields? *Vision Research*, 46:2901–2913, 2006.

[73] D Hansel, G. Mato, and C. Meunier. Synchrony in excitatory neural networks. *Neural Computation*, 7(2):307–337, Apr 1995.

[74] Bertil Hille. *Ion channels of excitable membranes. 3rd Edition*. Sinauer Associates, Inc., Sunderland, MA, 2001.

[75] Michael Hines. Efficient computation of branched nerve equations. *International Journal of Bio-Medical Computing*, 15(1):69–76, 1984.

[76] A.L. Hodgkin and A.F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *Journal of Physiology*, 117(4):500–44, 1952.

[77] Alois Hofbauer and José Campos-Ortega. Proliferation pattern and early differentiation of the optic lobes in *Drosophila melanogaster*. *Roux's Archives of Developmental Biology*, 198:264–274, 1990. doi: 10.1007/BF00377393.

[78] William .C. Hoffman. The Lie algebra of visual perception. *Journal of Mathematical Psychology*, 3(1):65–98, 1966. doi: 10.1016/0022-2496(66)90005-8.

[79] David H. Hubel and Torsten N. Wiesel. Receptive field, binocular interaction and functional architecture in the cat's visual cortex. *Journal of Physiology*, 160(1):106–154, 1962.

[80] Katsuhiko Inoue. 3D waltz of the flowers, 2009. URL `http://www.youtube.com/watch?v=0GcLW0g_c1s`.

[81] Eugene M Izhikevich. *Dynamical Systems in Neuroscience: The Geometry of Excitability and Bursting*. MIT Press, Cambridge, MA, 2007.

[82] Gilad A. Jocobson, Kamran Diba, Anat Yaron-Jakoubovitch, Yasmin Oz, Christof Koch, Idan Segev, and Yosef Yarom. Subthreshold voltage noise of rat neocortical pyramidal neurones. *Journal of Physiology*, 564(1):145–160, 2005. doi: 10.1113/jphysiol.2004.080903.

[83] Juson P. Jones and Larry A. Palmer. An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex. *Journal of Neurophysiology*, 58(6):1233–1258, 1987.

[84] J. B. Jonston. Thermal agitation of electricity in conductors. *Physics Review*, 29: 367–368, 1927.

[85] B. Katz. The Croonian lecture: The transmission of impulses from nerve to muscle, and the subcellular unit of synaptic action. *Proceedings of the Royal Society of London. Series B, Biological Sciences*, 155(961):455–477, 1962.

[86] Christoph Kayser, Nikos K. Logothetis, and Stefano Panzeri. Millisecond encoding precision of auditory cortex neurons. *Proceedings of the National Academy of Science of the United States of America*, 107(39):16976–16981, 2010. doi: 10.1073/pnas.1012656107.

[87] Justin Keat, Pamela Reinagel, R. Clay Reid, and Markus Meister. Predicting every spike: A model for the responses of visual neurons. *Neuron*, 30:803–817, 2001. doi: 10.1016/S0896-6273(01)00322-1.

[88] Anmo J. Kim and Aurel A. Lazar. Recovery of stimuli encoded with a hodgkin-huxley neuron using conditional prcs. In Nathan W. Schultheiss, Astrid A. Prinz, and Robert J. Butera, editors, *Phase Response Curves in Neuroscience*, chapter 11, pages 257–277. Springer, New York, NY, 2012. doi: 10.1007/978-1-4614-0739-3_11.

[89] Anmo J. Kim, Aurel A. Lazar, and Yevgeniy B. Slutskiy. System identification of Drosophila olfactory sensory neurons. *Journal of Computational Neuroscience*, 30(1): 143–161, February 2011. ISSN 0929-5313. doi: 10.1007/s10827-010-0265-0.

[90] Peter E. Kloeden and Eckhard Platen. *Numerical Solution of Stochastic Differential Equations*. Springer Berlin Heidelberg, 1992.

[91] Dwight J. Kravitz, Latrice D. Vinson, and Chris I. Baker. How position dependent is visual object recognition? *Trends in cognitive sciences*, 12(3):114–122, 2008. doi: 10.1016/j.tics.2007.12.006.

[92] Jens Kremkow, Jianzhong Jin, Stanley J. Komban, Yushi Wang, Reza Lashgari, Xiaobing Li, Michael Jansen, Qasim Zaidi, and Jose-Manuel Alonso. Neuronal nonlinearity explains greater visual spatial resolution for darks than lights. *Proceedings of the National Academy of Science of the United States of America*, 111(8):3170–3175, 2014. doi: 10.1073/pnas.1310442111.

[93] Stephen W. Kuffler. Discharge patterns and functional organization of mammalian retina. *Journal of Neurophysiology*, 16:37–68, 1953.

[94] Aurel A. Lazar. A simple model of spike processing. *Neurocomputing*, 69:1081–1085, 2006. doi: 10.1016/j.neucom.2005.12.050.

[95] Aurel A. Lazar. Population encoding with hodgkin-huxley neurons. *IEEE Transactions on Information Theory*, 56(2):821–837, February 2010. doi: 10.1109/TIT.2009.2037040. Special Issue on Molecular Biology and Neuroscience.

[96] Aurel A. Lazar and Eftychios A. Pnevmatikakis. Faithful representation of stimuli with a population of integrate-and-fire neurons. *Neural Computation*, 20(11):125–146, 2008. doi: 10.1162/neco.2008.06-07-559.

[97] Aurel A. Lazar and Eftychios A. Pnevmatikakis. Reconstruction of sensory stimuli encoded with integrate-and-fire neurons with random thresholds. *EURASIP Journal on Advances in Signal Processing*, Volume 2009:682930, 2009. doi: 10.1155/2009/682930. Special Issue on Statistical Signal Processing in Neuroscience.

[98] Aurel A. Lazar and Eftychios A. Pnevmatikakis. Video time encoding machines. *IEEE Transactions on Neural Networks*, 22(3):461–473, March 2011. doi: 10.1109/TNN.2010.2103323.

[99] Aurel A. Lazar and Eftychios A. Pnevmatikakis. Encoding of multivariate stimuli with MIMO neural circuits. In *Proceedings of the ISIT 2011*, Saint Petersburg, Russia, 2011. IEEE.

[100] Aurel A Lazar and Yevgeniy Slutskiy. Multisensory encoding, decoding, and identification. In C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3183–3191. Curran Associates, Inc., Lake Tahoe, NV, 2013.

[101] Aurel A. Lazar and Yevgeniy B. Slutskiy. Identifying dendritic processing. *Advances in Neural Information Processing Systems*, 23:1261–1269, 2010.

[102] Aurel A. Lazar and Yevgeniy B. Slutskiy. Channel identification machines. *Journal of Computational Intelligence and Neuroscience*, 2012:1–20, Jul 2012. doi: 10.1155/2012/209590.

[103] Aurel A. Lazar and Yevgeniy B. Slutskiy. Channel identification machines for multidimensional receptive fields. *Frontiers in Computational Neuroscience*, 8:117, 2014. doi: 10.3389/fncom.2014.00117.

[104] Aurel A. Lazar and Yevgeniy B. Slutskiy. Functional identification of spike-processing neural circuits. *Neural Computation*, 26(2):264–305, 2014. doi: 10.1162/NECO_a_00543.

[105] Aurel A. Lazar and Yevgeniy B. Slutskiy. Spiking neural circuits with dendritic stimulus processors: Encoding, decoding, and identification in reproducing kernel hilbert spaces. *Journal of Computational Neuroscience*, 38(1):1–24, 2015. doi: 10.1007/s10827-014-0522-8.

[106] Aurel A. Lazar and László Tóth. Perfect recovery and sensitivity analysis of time encoded bandlimited signals. *IEEE Transactions on Circuits and Systems-I: Regular Papers*, 51(10):2060–2073, 2004. doi: 10.1109/TCSI.2004.835026.

[107] Aurel A. Lazar and Chung-Heng Yeh. Functional identification of an antennal lobe dm4 projection neuron of the fruit fly. In *Abstracts from the Twenty Third Annual Computational Neuroscience Meeting: CNS*2014*, Québec City, Canada, July 2014. doi: 10.1186/1471-2202-15-S1-P5.

[108] Aurel A. Lazar and Yiyin Zhou. Massively parallel neural encoding and decoding of visual stimuli. *Neural Networks*, 32:303–312, 2012. doi: 10.1016/j.neunet.2012.02.007. Special Issue: IJCNN 2011.

[109] Aurel A. Lazar and Yiyin Zhou. Volterra dendritic stimulus processors and biophysical spike generators with intrinsic noise sources. *Frontiers in Computational Neuroscience*, 8:95, 2014. doi: 10.3389/fncom.2014.00095.

[110] Aurel A. Lazar and Yiyin Zhou. Reconstructing natural visual scenes from spike times. *Proceedings of the IEEE*, 102(10):1500–1519, 2014. doi: 10.1109/JPROC.2014.2346465.

[111] Aurel A. Lazar, Ernö K. Simonyi, and László Tóth. An overcomplete stitching algorithm for time decoding machines. *IEEE Transactions on Circuits and Systems-I: Regular Papers*, 55(9):2619–2630, October 2008. doi: 10.1109/TCSI.2008.920982.

[112] Aurel A. Lazar, Eftychios A. Pnevmatikakis, and Yiyin Zhou. Encoding natural scenes with neural circuits with random thresholds. *Vision Research*, 50(22):2200–2212, October 2010. doi: 10.1016/j.visres.2010.03.015. Special Issue on Mathematical Models of Visual Coding.

[113] Aurel A. Lazar, Wenze Li, Nikul H. Ukani, Chung-Heng Yeh, and Yiyin Zhou. Neural circuit abstractions in the fruit fly brain. *Society for Neuroscience Abstracts*, 2013.

[114] Aurel A. Lazar, Eftychios A. Pnevmatikakis, and Yiyin Zhou. The power of connectivity: Identity preserving transformations on visual streams in the spike domain. *Neural Networks*, 44:22–35, 2013. doi: 10.1016/j.neunet.2013.02.013.

[115] Aurel A. Lazar, Nikul H. Ukani, Chung-Heng Yeh, and Yiyin Zhou. A parallel programming model of local processing units in the fruit fly brain. *Frontiers in Neuroinformatics. Conference Abstract: Neuroinformatics*, (24), 2014. doi: 10.3389/conf.fninf.2014.18.00024.

[116] Aurel A. Lazar, Nikul H. Ukani, and Yiyin Zhou. The cartridge: A canonical neural circuit abstraction of the lamina neuropil construction and composition rules. *Neurokernel RFC #2*, 2014. doi: 10.5281/zenodo.11856.

[117] Aurel A. Lazar, Konstantinos Psychas, Nikul H. Ukani, and Yiyin Zhou. Retina of the fruit fly eyes: A detailed simulation model. *BMC Neuroscience 2015*, 14(Suppl 1): P301, 2015.

[118] Aurel A. Lazar, Yevgeniy B. Slutskiy, and Yiyin Zhou. Massively parallel neural circuits for stereoscopic color vision: Encoding, decoding and identification. *Neural Networks*, 63:254–271, 2015. doi: 10.1016/j.neunet.2014.10.014.

[119] Aurel A. Lazar, Nikul H. Ukani, and Yiyin Zhou. Functional identification of complex cells from spike times and the decoding of visual stimuli. *BMC Neuroscience 2015*, 14 (Suppl 1):P300, 2015.

[120] Tai Sing Lee. Image representation using 2d gabor wavelets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(10):959–971, 1996. doi: 10.1109/34.541406.

[121] Patrick Lichtsteiner, Christoph Posch, and Tobi Delbruck. A 128×128 dB, 15 μs latency asynchronous temporal contrast vision sensor. *IEEE Journal Of Solid-State Circuits*, 43(2):566–576, 2008. doi: 10.1109/JSSC.2007.914337.

[122] Daniele Linaro, Marco Storace, and Michele Giugliano. Accurate and fast simulatioin of channel noise in conductance-based model neurons by diffusion approximation. *PLoS Computational Biology*, 7(3):e1001102, 2011. doi: 10.1371/journal.pcbi.1001102.

[123] N. K. Logothetis and D. L. Sheinberg. Visual object recognition. *Annual Review of Neuroscience*, 19(1):577–621, 1996. doi: 10.1146/annurev.ne.19.030196.003045.

[124] Michael London and Michael Häusser. Dendritic computation. *Annual Review Neuroscience*, 28:503–532, 2005. doi: 10.1146/annurev.neuro.28.061604.135703.

[125] Henri Lorach, Ryad Benosman, Olivier Marre, Sio-Hoi Ieng, José A. Sahel, and Serge Picaud. Artificial retina: the multichannel processing of the mammalian retina achieved with a neuromorphic asynchronous light acquisition device. *Journal of Neural Engineering*, 9:066004, 2012. doi: 10.1088/1741-2560/9/6/066004.

[126] Ude Lu, Dong Song, and Theodore W. Berger. Nonlinear dynamic modeling of synaptically driven single hippocampal neuron intracelluar activity. *IEEE Transactions on Biomedical Engineering*, 58(5):1303–1313, 2011. doi: 10.1109/TBME.2011.2105870.

[127] Brian N. Lundstrom, Sungho Hong, Matthew H. Higgs, and Adrienne L. Fairhall. Two computational regimes of a single compartment neuron separated by a planar boundary in conductance space. *Neural Computation*, 20(5):1239–1260, 2008. doi: 10.1162/neco.2007.05-07-536.

[128] Amit Manwani and Christof Koch. Detecting and estimating signals in noisy cable structures, i: Neuronal noise sources. *Neural computation*, 11:1797–1829, 1999. doi: 10.1162/089976699300015972.

[129] David Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. MIT Press, Cambridge, MA, 2010.

[130] R. H. Masland. The fundamental plan of the retina. *Nature neuroscience*, 4(9):877–886, 2001. doi: 10.1038/nn0901-877.

[131] Richard H. Masland. The neuronal organization of the retina. *Neuron*, 76(18):266–280, 2012. doi: 10.1016/j.neuron.2012.10.002.

[132] Nicolas Y. Masse, Glenn C. Turner, and Gregory S.X.E. Jefferis. Olfactory information processing in *Drosophila*. *Current Biology*, 19(16):R700–R713, 2009. doi: 10.1016/j.cub.2009.06.026.

[133] Mark D. McDonnell and Lawrence M. Ward. The benefits of noise in neural systems: bridging theory and experiment. *Natural Review Neuroscience*, 12:415–425, 2011. doi: 10.1038/nrn3061.

[134] Claude Meunier and Idan Segev. Playing the devil's advocate: is the Hodgkin-Huxley model useful? *TRENDS in Neurosciences*, 25(11):558–563, 2002. doi: 10.1016/S0166-2236(02)02278-6.

[135] C. Morris and H. Lecar. Voltage oscillations in the barnacle giant muscle fiber. *Biophysical Journal*, 35(1):193–213, Jul 1981.

[136] J. A. Movshon, I. D. Thompson, and David J. Tolhurst. Spatial summation in the receptive fields of simple cells in the cat's striate cortex. *Journal of Physiology*, 283: 53–77, 1978.

[137] E. Nattel and Y. Yeshurun. An efficient data structure for feature extraction in a foveated environment. In *Biologically Motivated Computer Vision*, pages 139–165. Springer, 2000.

[138] Erwin Neher and Bert Sakmann. Single-channel currents recorded from membrane of denervated frog muscle fibres. *Nature*, 260:799–802, 1976. doi: 10.1038/260799a0.

[139] Aljoscha Nern, Barret D. Pfeiffer, and Gerald M. Rubin. Optimized tools for multicolor stochastic labeling reveal diverse stereotyped cell arraangements in the fly visual system. *PNAS*, 112(22):E2967–76, 2015. doi: 10.1073/pnas.1506763112.

[140] Bruno A. Olshausen and David J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–609, 1996. doi: 10.1038/381607a0.

[141] Bruno A. Olshausen, Charles H. Anderson, and David C. Van Essen. A neurobiological model of visual attention and invarinant pattern recognition based on dynamic routing of information. *Journal of Neuroscience*, 13(11):4700–4719, 1993.

[142] Alan V. Oppenheim, Ronald W. Schafer, and John R. Buck. *Discrete-Time Signal Processing*. Prentice Hall, 2nd edition, 1999.

[143] Patricio Orio and Daniel Soudry. Simple, fast and accurate implementation of the diffusion approximation algorithm for stochastic ion channels with multiple states. *PLoS One*, 7(5):e36670, 2012. doi: 10.1371/journal.pone.0036670.

[144] Srdjan Ostojic and Nicolas Brunel. From spiking neuron models to linear-nonlinaer models. *PLoS Computational Biology*, 7(1):e1001056, 2011. doi: 10.1371/journal.pcbi.1001056.

[145] Liam Paninski. Maximum likelihood estimation of cascade point-process neural encoding models. *Network: Computation in Neural Systems*, 15:243–262, 2004.

[146] Liam Paninski, Jonathan W. Pillow, and Eero P. Simoncelli. Maximum likelihood estimation of a stochastic integrate-and-fire neural encoding model. *Neural Computation*, 16:2533–2561, 2004. doi: 10.1162/0899766042321797.

[147] Jonathan W. Pillow and Eero P. Simoncelli. Dimensionality reduction in neural models: an information-theoretic generalization of spike-triggered average and covariance analysis. *Journal of Vision*, 6(4):414–428, 2006. doi: 10.1167/6.4.9.

[148] Jonathan W. Pillow, Liam Paninski, V. J. Uzzell, Eero P. Simoncelli, and E. J. Chichilnisky. Prediction and decoding of retinal ganglion cell responses with a probabilistic spiking model. *Journal of Neuroscience*, 25(47):11003–11013, 2005. doi: 10.1523/JNEUROSCI.3305-05.2005.

[149] Jonathan W. Pillow, Jonathon Shlens, Liam Paninski, Alexander Sher, Alan M. Litke, E. J. Chichilnisky, and Eero P. Simoncelli. Spatio-temporal correlations and visual signaling in a complete neuronal population. *Nature*, 454:995–999, 2008. doi: 10.1038/nature07140.

[150] Jonathan W. Pillow, Yashar Ahmadian, and Liam Paninski. Model-based decoding, information estimation, and change-point detection techniques for multineuron spike trains. *Neural Computation*, 23:1–45, 2011. doi: 10.1162/NECO_a_00058.

[151] Eftychios A. Pnevmatikakis. *Spikes as Projections: Representation and Processing of Sensory Stimuli in the Time Domain*. PhD thesis, Columbia University, 2009.

[152] Tomaso Poggio and Shimon Edelman. A network that learns to recognize three-dimensional objects. *Nature*, 343:263–266, 1990. doi: 10.1038/343263a0.

[153] Uri Polat and Dov Sagi. Lateral interactions between spatial channels: Suppression and facilitation revealed by lateral masking experiments. *Vision Research*, 33(7):993–999, 1993. doi: 10.1016/0042-6989(93)90081-7.

[154] C. Posch. Bio-inspired vision. *Journal of Instrumentation*, 7(1):C01054, 2012. doi: 10.1088/1748-0221/7/01/C01054.

[155] Ning Qian. Binocular disparity and the perception of depth. *Neuron*, 18:359–368, 1997. doi: 10.1016/S0896-6273(00)81238-6.

[156] Ning Qian and Yu-Dong Zhu. Physiological computation of binocular disparity. *Vision Research*, 37(13):1811–1827, 1997.

[157] Minh Ha Quang, Sung Ha Kang, and Triet M. Le. Image and video colorization using vector-valued reproducing kernel hilbert spaces. *Journal of Mathematical Imaging and Vision*, 37(1):49–65, 2010. doi: 10.1007/s10851-010-0192-8.

[158] Charles P. Ratliff, Bart G. Borghuis, Yen-Hong Kao, Peter Sterling, and Vijay Balasubramanian. Retina is structured to process an excess of darkness in natural scenes. *Proceedings of the National Academy of Science of the United States of America*, 107 (40):17368–17373, 2010. doi: 10.1073/pnas.1005846107.

[159] R. C. Reid, R. E. Soodak, and R. M. Shapley. Directional selectivity and spatiotemporal structure of receptive fields of simple cells in cat striate cortex. *Journal of Neurophysiology*, 66:505–529, August 1991.

[160] Fred Rieke and Michael E. Rudd. The challenges natural images pose for visual adaptation. *Neuron*, 64:605–616, 2009. doi: 10.1016/j.neuron.2009.11.028.

[161] Fred Rieke, David Warland, Rob de Ruyter van Steveninck, and William Bialek. *Spikes: Exploring the Neural Code*. The MIT Press, Cambridge, MA, 1999.

[162] D. L. Ringach and M. J. Hawken. Orientation selectivity in macaque v1: diversity and laminar dependence. *Journal of Neuroscience*, 22:5639–5651, 2002.

[163] Marta Rivera-Alba, Shiv N. Vitaladevuni, Yuriy Mischenko, Zhiyuan Lu, Shin-ya Takemura, Lou Scheffer, Ian A. Meinertzhagen, Dmitri B. Chklovskii, and Gonzalo G. de Polavieja. Wiring economy and volume exclusion determine neuronal placement in the *Drosophila* brain. *Current Biology*, 2011. doi: 10.1016/j.cub.2011.10.022.

[164] R. W. Rodieck. Quantitative analysis of cat retinal ganglion cell response to visual stimuli. *Vision Research*, 5(11):583–601, 1965. doi: 10.1016/0042-6989(65)90033-7.

[165] Wilson J. Rugh. *Nonlinear System Theory: Volterra/Wiener Approach*. Johns Hopkins University Press, Baltimore, MD, 1981.

[166] R Russell and P Sinha. Real-world face recognition: the importance of surface reflectance properties. *Perception*, 36(9), 2007.

[167] M Sarangdhar and C. Kambhampati. A numerical model for Hodgkin-Huxley neural stimulus reconstruction. *IAENG International Journal of Computer Science*, 38(1): 89–94, 2011.

[168] Shannon Saszik and Steven H. DeVries. A mammalian retinal bipolar cell uses both graded changes in membrane voltage and all-or-Nothing na^+ spikes to encode light. *Journal of Neuroscience*, 32(1):297–307, 2012. doi: 10.1523/JNEUROSCI.2739-08.2012.

[169] Odelia Schwartz, Jonathan W. Pillow, Nicole C. Rust, and Eero P. Simoncelli. Spike-triggered neural characterization. *Journal of Vision*, 6:484–207, 2006. doi: 10.1167/6.4.13.

[170] C. E. Shannon. Communication in the presence of noise. *Proc. Institue of Radio Engineerings*, 37(1):10–21, 1949.

[171] R. M. Shapley and J. D. Victor. The effect of contrast on the transfer properties of cat retinal ganglion cells. *Journal of Physiology*, 285:275–298, 1978.

[172] Robert Shapley and Christina Enroth-Cugell. Visual adaptation and retinal gain controls. *Progress in Retinal Research*, 3:263–346, 1984.

[173] E.P. Simoncelli, W.T. Freeman, E.H. Adelson, and D.J. Heeger. Shiftable multiscale transform. *IEEE Transactions on Information Theory*, 38(2):587–607, March 1992. doi: 10.1109/18.119725.

[174] Benjamin Sivyer and Stephen R. Williams. Direction selectivity is computed by active dendritic integration in retinal ganglion cells. *Nature Neuroscience*, 16:1848–1856, 2013. doi: 10.1038/nn.3565.

[175] Samuel G. Solomon and Peter Lennie. The machinery of colour vision. *Nature Review Neuroscience*, 8:276–286, 2007. doi: 10.1038/nrn2094.

[176] Zhuoyi Song, Marten Postma, Stephen A. Billings, Daniel Coca, Roger C. Hardie, and Mikko Juusola. Stochastic, adaptive sampling of information by microvilli in fly photoreceptors. *Current Biology*, 22:1371–1380, 2012. doi: 10.1016/j.cub.2012.05.047.

[177] Garrett Stanley, Fei F. Li, and Yang Dan. Reconstruction of natural scenes from ensemble responses in the lateral geniculate nucleus. *Journal of Neuroscience*, 19(18): 8036–8042, 1999.

[178] Peter Sterling and Jonathan B. Demb. Retina. In Gordon M. Shepherd, editor, *The Synaptic Organization of the Brain*. Oxford University Press, New York, 5th ed. edition, 2004. doi: 10.1093/acprof:oso/9780195159561.003.0006.

[179] Greg J. Stuart and Michael Häusser. Dendritic coincidence detection of EPSPs and action potentials. *Nature Neuroscience*, 4:63–71, 2001. doi: 10.1038/82910.

[180] Shin-ya Takemura, Arjun Bharioke, Zhiyuan Lu, Aljoscha Nern, Shiv Vitaladevuni, Patricia K. Rivlin, William T. Katz, Donald J. Olbris, Stephen M. Plaza, Philip Winston, Ting Zhao, Jane Anne Horne, Richard D. Fetter, Satoko Takemura, Katerina Blazek, Lei-Ann Chang, Omotara Ogundeyi, Mathew A. Saunders, Victor Shaprio, Christopher Sigmund, Gerald M. Rubin, Louis K. Scheffer, Ian A. Meinertzhagen, and Dmitri B. Chklovskii. A visual motion detection circuit suggested by *Drosophila* connectomics. *Nature*, 500:175–181, 2013.

[181] Michael J. Tarr and Steven Pinker. Mental rotation and orientation-dependence in shape recognition. *Cognitive Psychology*, 21:233–282, 1989. doi: 10.1016/0010-0285(89)90009-1.

[182] Frédéric E. Theunissen, Kamal Sen, and Allison J. Doupe. The receptive field of the primate P retinal ganglion cell, II: Nonlinear dynapmics. *Journal of Neuroscience*, 20 (6):2315–2331, 2000. doi: 10.1017/S0952523800008865.

[183] R. H. Tutuncu, K. C. Toh, and M. J. Todd. Solving semidefinite-quadratic-linear programs using sdpt3. *Mathematical Programming Ser. B*, 95:189–217, 2003.

[184] Shimon Ullman. Aligning pictorial descriptions: An approach to object recogition. *Cognition*, 32:193–254, 1989.

[185] Michael Unser. Sampling – 50 years after shannon. *Proceedings of the IEEE*, 88(4): 569–587, 2000. doi: 10.1109/5.843002.

[186] Rufin Van Rullen and Simon J. Thorpe. Rate coding versus termporal order coding: What the retinal ganglion cells tell the visual cortex. *Neural Computation*, 13:1255–1283, 2001. doi: 10.1162/08997660152002852.

[187] Vito Volterra. *Theory of Functionals and of Integral and Integro-Differential Equations*. Dover Publications, New York, 1930.

[188] G. Wahba. *Spline Models for Observational Data*. Society for Industrial Mathematics, 1990.

[189] Brian A. Wandell. *Foundations of Vision*. Sinauer, Sunderland, MA, 1995.

[190] Z. Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. doi: 10.1109/TIP.2003.819861.

[191] Barry Wark, Brain Nils Lundstrom, and Adrienne L. Fairhall. Sensory adpatation. *Current Opinion in Neurobiology*, 17:423–429, 2007. doi: 10.1016/j.conb.2007.07.001.

[192] David K. Warland, Pamela Reinagel, and Markus Meister. Decoding visual information from a population of retinal ganglion cells. *Journal of Neurophysiology*, 78: 2336–2350, 1997.

[193] C. Weber and J. Triesch. Implementations and implications of foveated vision. *Recent Patents on Computer Science*, 2(1):75–85, 2009. doi: 10.2174/2213275910902010075.

[194] Frank S. Werblin. The retinal hypercircuit: a repeating synaptic interactive motif underlying visual function. *Journal of Physiology*, 589(15):3691–3702, 2011. doi: 10.1113/jphysiol.2011.210617.

[195] G. Westheimer and B.L. Beard. Orientation dependency for foveal line stimuli: detection and intensity discrimination, resolution, orientation discrimination and vernier acuity. *Vision research*, 38(8):1097–1103, 1998. doi: 10.1016/S0042-6989(97)00248-4.

[196] John A. White, Ruby Klink, Angel Alonso, and Alan R. Kay. Noise from voltage-gated ion channels may influence neuronal dynamics in the entorhinal cortex. *Journal of Neurophysiology*, 80:262–269, 1998.

[197] John A. White, Jay T. Rubinstein, and Alan R. Kay. Channel noise in neurons. *TRENDS in Neurosciences*, 23(3):131–137, 2000. doi: 10.1016/S0166-2236(99)01521-0.

[198] Adrien Wohrer and Pierre Kornprobst. Virtual Retina: A biological retina model and simulator, with contrast gain control. *Journal of Computational Neuroscience*, 26:219–249, 2009. doi: 10.1007/s10827-008-0108-4.

[199] Philipp Wolfrum and Christoph von der Malsburg. What is the optimal architecture for visual information routing? *Neural Computation*, 19:3293–3309, 2007. doi: 10.1162/neco.2007.19.12.3293.

[200] Ning-long Xu, Mark T. Harnett, Stephen R. Williams, Daniel Huber, Daniel H. O'Connor, Karel Svoboda, and Jeffrey C. Magee. Nonlinear dendritic integration of sensory and motor input during an active sensing task. *Nature*, 492:247–251, 2012. doi: 10.1038/nature11601.

[201] Yosef Yarom and Jorn Hounsgaard. Voltage fluctuations in neurons: Signal or noise? *Physiology Review*, 91:917–929, 2011. doi: 10.1152/physrev.00019.2010.

[202] Keisuke Yonehara, Karl Farrow, Alexander Ghanem, Daniel Hillier, Kamill Balint, Miguel Teixeira, Josephine Jüttner, Masaharu Noda, Rachael L. Neve, Karl-Klaus Conzelmann, and Botond Roska. The first stage of cardinal direction selectivity is localized to the dendrites of retinal ganglion cells. *Neuron*, 79:1078–1085, 2013. doi: 10.1016/j.neuron.2013.08.005.

[203] Kareem A. Zaghloul and Kwabena Boahen. A silicon retina that reproduces signals in the optic nerve. *Journal of Neural Engineering*, 3:257–267, 2006. doi: 10.1088/1741-2560/3/4/002.

[204] Danke Zhang, Yuanqing Li, Malte J. Rasch, and Si Wu. Nonlinear multiplicative dendritic integration in neuron and network models. *Frontiers in Computational Neuroscience*, 7:56, 2013. doi: 10.3389/fncom.2013.00056.

[205] Yu-Dong Zhu and Ning Qian. Binocular receptive field models, disparity tuning, and characteristic disparity. *Neural Computation*, 8:1611–1641, 1996. doi: 10.1162/neco.1996.8.8.1611.

Appendix A

The Hodgkin-Huxley Neuron Model

A.1 The Hodgkin-Huxley Equations

It was shown that the excitable membrane of a squid giant axon can be represented as the equivalent circuit shown in Figure A.1 [76]. The circuit consists of a capacitor, a voltage-gated sodium channel, a voltage-gated potassium channel, and a leak channel.

The above equivalent circuit can be expressed by a system of differential equations. We consider Hodgkin-Huxley neuron model with standard parameters, described by the non-linear differential equations:

$$\begin{aligned} C\frac{dV}{dt} &= -\overline{g}_{Na}m^3h(V-E_{Na}) - \overline{g}_K n^4(V-E_K) - \overline{g}_L(V-E_L) + I \\ \frac{dn}{dt} &= \alpha_n(V)(1-n) - \beta_n(V)n \\ \frac{dm}{dt} &= \alpha_m(V)(1-m) - \beta_m(V)m \\ \frac{dh}{dt} &= \alpha_h(V)(1-h) - \beta_h(V)h \end{aligned} \quad , \tag{A.1}$$

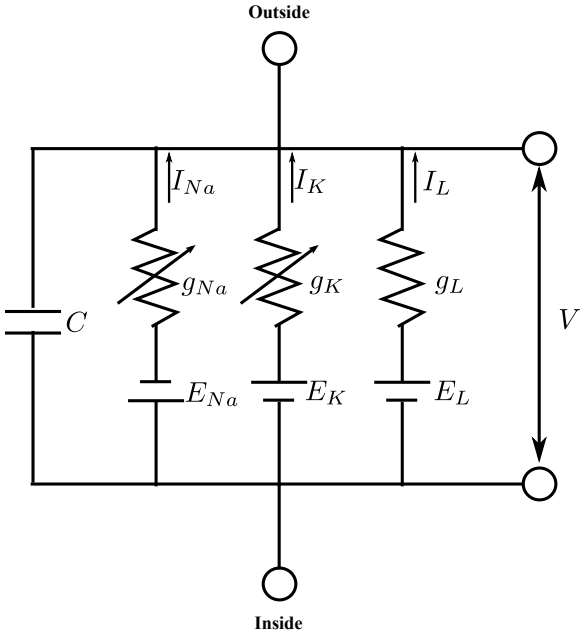


Figure A.1: Equivalent circuit of a Hodgkin-Huxley neuron membrane.

and

$$\begin{aligned}
\alpha_n(V) &= \frac{0.01(V+55)}{1-e^{-\frac{V+55}{10}}} \\
\beta_n(V) &= 0.125e^{-\frac{V+65}{80}} \\
\alpha_m(V) &= \frac{0.1(V+40)}{1-e^{-\frac{V+40}{10}}} \\
\beta_m(V) &= 4e^{-\frac{V+65}{18}} \\
\alpha_h(V) &= 0.07e^{-\frac{V+65}{20}} \\
\beta_h(V) &= \frac{1}{1+e^{-\frac{V+35}{18}}}
\end{aligned} \quad , \tag{A.2}$$

where V is the membrane potential, n, m, h are the gating variables, and I is the bias current. The latter is assumed to be large enough to induce periodic spiking. Therefore, the Hodgkin-Huxley neuron considered here is a periodically spiking neuron with period $T(I)$, where $T = T(I)$ maps the bias current I into the period of spiking T. The function T is shown in Figure A.2. In other words, $T(I)$ is closely associated with the $f - I$ curve typically seen in the literature. Without loss of generality, we will assume for simplicity that $C = 1\mu F/cm^2$. Since the Hodgkin-Huxley neuron is periodically spiking, it has a well-defined

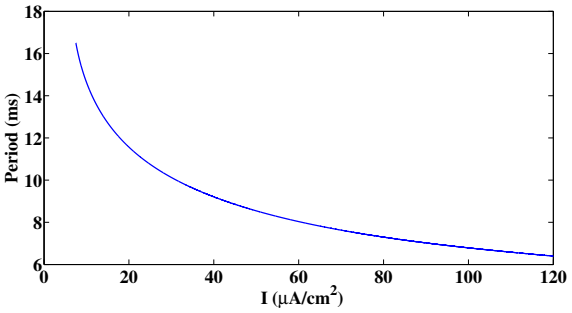


Figure A.2: Period of oscillation when Hodgkin-Huxley neuron is subject to a constant current I.

PRC $\boldsymbol{\psi}(t, I) = [\psi_1(t, I), \psi_2(t, I), \psi_3(t, I), \psi_4(t, I)]^{\mathsf{T}}$, where $\psi_1(t, I), \psi_2(t, I), \psi_3(t, I), \psi_4(t, I)$ are the PRCs associated with the component states V, n, m, h, respectively.

A.2 Phase Response Curves

Many conductance-based neuron models, including the Hodgkin-Huxley neuron model, exhibit asymptotically stable limit cycles for a range of bias current I. Assuming that a Hodgkin-Huxley neuron is periodically spiking, *i.e.*, the state of the neuron is moving on a limit cycle, then the state can be described simply on a one-dimensional phase space.

Let a limit cycle associated with bias current I be denoted as $\mathbf{x}^o(t)$, the T-periodic solution to (A.1) with initial condition at $\mathbf{x}^o(0)$. Obviously, $\mathbf{x}^o(t), t \in [0, T)$ is bijective to the set of phase $\theta = [0, T)$ as $\mathbf{x}^o(t_1) \neq \mathbf{x}^o(t_2), \forall t_1, t_2 \in [0, T), t_1 \neq t_2$. Here, we use unnormalized phase which takes a range from 0 to T rather than using normalized phase that ranges from 0 to 1. Therefore, the states on the limit cycle can be fully describe by the phase.

Consider now a small perturbation of the state that can be applied by introducing a brief external current source. The effect of the perturbation is the deviation of trajectory from the limit cycle when the external current is introduced. As the brief period of external current ends, the state of the neuron is attracted by the asymptotically stable limit cycles and it gradually settles back to the limit cycle. However, when the state of the neuron settles back

onto the limit cycle, its phase on the limit cycle may be delayed or advanced compare to the case if the neuron were allowed to continue moving on the limit cycle unperturbed. The amount of shift in phase due to a perturbation at phase θ is called a phase shift. Therefore, the perturbation has two effects. First, it introduces an orbital deviation that drags the states out of the limit cycle. Second, it introduces a phase shift that advance or delay the progression of state on the limit cycle.

The phase shift when the neuron is at phase θ and subject to an impulse is called the *phase response curve* (PRC). A PRC is the linearization of neural response around a limit cycle in the phase space. It indicates how much phase shift is applied to the system when it is subject to an external current source.

PRC associated with a limit cycle can be obtained as the solution to the adjoint system of the linearized system around the limit cycle. That is,

$$\frac{d\boldsymbol{\psi}}{dt}(t, I) = -\mathbf{A}(t, I)^{\mathsf{T}}\boldsymbol{\psi}(t, I) \tag{A.3}$$

with $\boldsymbol{\psi}(0, I)\mathbf{f}(\mathbf{x}^o(0, I)) = 1$, where $\mathbf{A}(t, I) = \mathcal{D}(\mathbf{f}(\mathbf{x}^o(t, I)))^{\mathsf{T}}$, the transposed Jacobian of $\mathbf{f}$ at point $\mathbf{x}^o(t, I)$ on the limit cycle [81].

A.3 Hodgkin-Huxley Neuron with Channel Noise

A.3.1 Hodgkin-Huxley Equations with Subunit Noise

We first consider a stochastic model where subunit noise is introduced to the Hodgkin-Huxley equations [67]. The stochastic model can be expressed as the following SDE (for clarity, neuron index is not shown)

$$d\mathbf{Y} = \mathbf{f}(\mathbf{Y}, I)dt + d\mathbf{Z}(t), \tag{A.4}$$

where $\mathbf{f}$ is defined as in (A.1) with additional normalization such that the unit of time is in seconds instead of milliseconds and the unit of voltage is in Volts instead of milivolts as

conventionally used. $\mathbf{Z}(t)$ takes the form

$$d\mathbf{Z}(t) = \begin{bmatrix} vdt + \sigma_1 dW_1 \\ \sigma_2 dW_2 \\ \sigma_3 dW_3 \\ \sigma_4 dW_4 \end{bmatrix}.$$

Here $W_n(t), n = 1, 2, 3, 4$ are independent standard Brownian motion processes and $\sigma_n, n = 1, 2, 3, 4$, are associated scaling factors. The first term $\sigma_1 dW_1$ models the noise introduced by DSPs, and the terms $\sigma_n dW_n, n = 2, 3, 4$ are known as subunit noise [67].

A.3.2 Hodgkin-Huxley Equations with Conductance Noise

We construct here stochastic ion channels using a model of conductance noise rather than the subunit noise used in (A.4) [67, 68]. This stochastic Hodgkin-Huxley system is simulated using the diffusion approximation of [143]. The system of SDEs can be expressed as (for clarity, neuron index is not shown)

$$d\mathbf{Y} = \mathbf{f}(\mathbf{Y}, I)dt + \mathbf{B}(\mathbf{Y})d\mathbf{Z}(t),$$

where Y has 14 state variables:

$$\mathbf{Y} = [V, N_0, N_1, N_2, N_3, N_4, M_0H_0, M_1H_0, M_2H_0, M_3H_0, M_0H_1, M_1H_1, M_2H_1, M_3H_1]^{\mathsf{T}},$$

where $N_i, i = 0, \cdots, 4$, denote the subunit states of potassium channels and $M_iH_j, i = 0, \cdots, 3, j = 0, 1$, denote the stochastic processes modeling the subunit states of sodium channels. The SDE is defined as:

$$\begin{aligned}
dV =& (-\overline{g}_{Na} M_3 H_1 (V - E_{Na}) - \overline{g}_K N_4 (V - E_K) - \overline{g}_L (V - E_L) + I + v)dt + \sigma_1 dW_1 \\
dN_0 =& (-4\alpha_n N_0 + \beta_n N_1)dt + \frac{1}{\sqrt{N_K}} \sqrt{4\alpha_n N_0 + \beta_n N_1} dW_2 \\
dN_1 =& (4\alpha_n N_0 - \beta_n N_1 - 3\alpha_n N_1 + 2\beta_n N_2)dt \\
& - \frac{1}{\sqrt{N_K}} \sqrt{4\alpha_n N_0 + \beta_n N_1} dW_2 + \frac{1}{\sqrt{N_K}} \sqrt{3\alpha_n N_1 + 2\beta_n N_2} dW_3
\end{aligned}$$

$$
\begin{aligned}
dN_2 =&(3\alpha_n N_1 - 2\beta_n N_2 - 2\alpha_n N_2 + 3\beta_n N_3)dt \\
&- \frac{1}{\sqrt{N_K}}\sqrt{3\alpha_n N_1 + 2\beta_n N_2}dW_3 + \frac{1}{\sqrt{N_K}}\sqrt{2\alpha_n N_2 + 3\beta_n N_3}dW_4 \\
dN_3 =&(2\alpha_n N_2 - 3\beta_n N_3 - \alpha_n N_3 + 4\beta_n N_4)dt \\
&- \frac{1}{\sqrt{N_K}}\sqrt{2\alpha_n N_2 + 3\beta_n N_3}dW_4 + \frac{1}{\sqrt{N_K}}\sqrt{\alpha_n N_3 + 4\beta_n N_4}dW_5 \\
dN_4 =&(\alpha_n N_3 - 4\beta_n N_4)dt - \frac{1}{\sqrt{N_K}}\sqrt{\alpha_n N_3 + 4\beta_n N_4}dW_5 \\
dM_0H_0 =&(-3\alpha_m M_0H_0 + \beta_m M_1H_0 - \alpha_h M_0H_0 + \beta_h M_0H_1)dt \\
&+ \frac{1}{\sqrt{N_{Na}}}\sqrt{3\alpha_m M_0H_0 + \beta_m M_1H_0}dW_6 + \frac{1}{\sqrt{N_{Na}}}\sqrt{\alpha_h M_0H_0 + \beta_h M_0H_1}dW_9 \\
dM_1H_0 =&(3\alpha_m M_0H_0 - \beta_m M_1H_0 - 2\alpha_m M_1H_0 + 2\beta_m M_2H_0 - \alpha_h M_1H_0 + \beta_h M_1H_1)dt \\
&- \frac{1}{\sqrt{N_{Na}}}\sqrt{3\alpha_m M_0H_0 + \beta_m M_1H_0}dW_6 + \frac{1}{\sqrt{N_{Na}}}\sqrt{2\alpha_m M_1H_0 + 2\beta_m M_2H_0}dW_7 \\
&+ \frac{1}{\sqrt{N_{Na}}}\sqrt{\alpha_h M_1H_0 + \beta_h M_1H_1}dW_{10} \\
dM_2H_0 =&(2\alpha_m M_1H_0 - 2\beta_m M_2H_0 - \alpha_m M_2H_0 + 3\beta_m M_3H_0 - \alpha_h M_2H_0 + \beta_h M_2H_1)dt \\
&- \frac{1}{\sqrt{N_{Na}}}\sqrt{2\alpha_m M_1H_0 + 2\beta_m M_2H_0}dW_7 + \frac{1}{\sqrt{N_{Na}}}\sqrt{\alpha_m M_2H_0 + 3\beta_m M_3H_0}dW_8 \\
&+ \frac{1}{\sqrt{N_{Na}}}\sqrt{\alpha_h M_2H_0 + \beta_h M_2H_1}dW_{11} \\
dM_3H_0 =&(\alpha_m M_2H_0 - 3\beta_m M_3H_0 - \alpha_h M_3H_0 + \beta_h M_3H_1)dt \\
&- \frac{1}{\sqrt{N_{Na}}}\sqrt{\alpha_m M_2H_0 + 3\beta_m M_3H_0}dW_8 + \frac{1}{\sqrt{N_{Na}}}\sqrt{\alpha_h M_3H_0 + \beta_h M_3H_1}dW_{12} \\
dM_0H_1 =&(-3\alpha_m M_0H_1 + \beta_m M_1H_1 + \alpha_h M_0H_0 - \beta_h M_0H_1)dt \\
&+ \frac{1}{\sqrt{N_{Na}}}\sqrt{3\alpha_m M_0H_1 + \beta_m M_1H_1}dW_{13} - \frac{1}{\sqrt{N_{Na}}}\sqrt{\alpha_h M_0H_0 + \beta_h M_0H_1}dW_9 \\
dM_1H_1 =&(3\alpha_m M_0H_1 - \beta_m M_1H_1 - 2\alpha_m M_1H_1 + 2\beta_m M_2H_1 + \alpha_h M_1H_0 - \beta_h M_1H_1)dt \\
&- \frac{1}{\sqrt{N_{Na}}}\sqrt{3\alpha_m M_0H_1 + \beta_m M_1H_1}dW_{13} + \frac{1}{\sqrt{N_{Na}}}\sqrt{2\alpha_m M_1H_1 + 2\beta_m M_2H_1}dW_{14} \\
&- \frac{1}{\sqrt{N_{Na}}}\sqrt{\alpha_h M_1H_0 + \beta_h M_1H_1}dW_{10}
\end{aligned}
$$

$$
\begin{aligned}
dM_2H_1 =&(2\alpha_m M_1H_1 - 2\beta_m M_2H_1 - \alpha_m M_2H_1 + 3\beta_m M_3H_1 + \alpha_h M_2H_0 - \beta_h M_2H_1)dt \\
&- \frac{1}{\sqrt{N_{Na}}}\sqrt{2\alpha_m M_1H_1 + 2\beta_m M_2H_1}dW_{14} + \frac{1}{\sqrt{N_{Na}}}\sqrt{\alpha_m M_2H_1 + 3\beta_m M_3H_1}dW_{15} \\
&- \frac{1}{\sqrt{N_{Na}}}\sqrt{\alpha_h M_2H_0 + \beta_h M_2H_1}dW_{11} \\
dM_3H_1 =&(\alpha_m M_2H_1 - 3\beta_m M_3H_1 + \alpha_h M_3H_0 - \beta_h M_3H_1)dt \\
&- \frac{1}{\sqrt{N_{Na}}}\sqrt{\alpha_m M_2H_1 + 3\beta_m M_3H_1}dW_{15} - \frac{1}{\sqrt{N_{Na}}}\sqrt{\alpha_h M_3H_0 + \beta_H M_3H_1}dW_{12}
\end{aligned}
$$

where $\alpha_n = \alpha_n(V), \beta_n = \beta_n(V), \alpha_m = \alpha_m(V), \beta_m = \beta_m(V), \alpha_h = \alpha_h(V), \beta_h = \beta_h(V)$ are defined as in the standard form (A.2), N_K, N_{Na} are the numbers of potassium and sodium channels, respectively, and $W_n, n = 1, \cdots, 15$ are independent Brownian motion processes.

Appendix B

Proof of Theorems and Equations

B.1 The Representer Theorem

Theorem 11. *Let $\mathcal{H}$ be an RKHS with reproducing kernel $K(s,t)$. Given bounded linear functionals $\mathcal{T}_k, k = 1, 2, \cdots, N$ and measurements $q_k, k = 1, 2, \cdots, N$, the minimizer $\hat{u}$ over the RKHS $\mathcal{H}$ of the regularized cost functional*

$$\lambda \|u\|_{\mathcal{H}}^2 + \frac{1}{N} \sum_{k=1}^{N} (\mathcal{T}_k u - q_k)^2$$

is of the form

$$\hat{u}(t) = \sum_{k=1}^{N} c_k \phi_k(t),$$

where $\phi_k(t) = \mathcal{T}_k K_t$, for all $k = 1, 2, \cdots, N$, and $K_t(\cdot) = K(\cdot, t)$.

Proof: By Riesz representation theorem, there exists a function $\phi_k \in \mathcal{H}$ such that

$$\mathcal{T}_k u = \langle u, \phi_k \rangle, \text{ for all } u \in \mathcal{H},$$

and

$$\phi_k(t) = \langle \phi_k, K_t \rangle = \mathcal{T}_k K_t.$$

Let $\mathcal{H}_0$ be a linear subspace of $\mathcal{H}$ defined by

$$\mathcal{H}_0 = \text{span}(\{\phi_k | k = 1, 2, \cdots, N\})$$

and $\mathcal{H}_0^\perp$ be a linear subspace of $\mathcal{H}$ defined by

$$\mathcal{H}_0^\perp = \{u \in \mathcal{H} | \mathcal{T}_k u = 0, k = 1, 2, \cdots, N\}.$$

For any $u \in \mathcal{H}_0^\perp$ and any $\sum_{k=1}^N c_k \phi_k \in \mathcal{H}_0$, we have

$$\langle u, \sum_{k=1}^N c_k \phi_k \rangle = \sum_{k=1}^N c_k \langle u, \phi_k \rangle = \sum_{k=1}^N c_k \mathcal{T}_k u = 0.$$

Since $\mathcal{H} = \mathcal{H}_0 \oplus \mathcal{H}_0^\perp$, u can be represented as $u = u_0 + u_0^\perp$ where $u_0 \in \mathcal{H}_0$ and $u_0^\perp \in \mathcal{H}_0^\perp$ are orthogonal. Therefore,

$$\|u_0 + u_0^\perp\|_{\mathcal{H}}^2 = \|u_0\|_{\mathcal{H}}^2 + \|u^\perp\|_{\mathcal{H}}^2.$$

Finally, since

$$\frac{1}{N}\sum_{k=1}^N (\mathcal{T}_k(u_0 + u_0^\perp - q_k)^2 = \frac{1}{N}\sum_{k=1}^N (\mathcal{T}_k u_0 - q_k)^2,$$

we have

$$\lambda\|u_0\|_{\mathcal{H}}^2 + \frac{1}{N}\sum_{k=1}^N (\mathcal{T}_k u_0 - q_k)^2 \leq \lambda\|u_0 + u_0^\perp\|_{\mathcal{H}}^2 + \frac{1}{N}\sum_{k=1}^N (\mathcal{T}_k(u_0 + u_0^\perp) - q_k)^2.$$

Clearly, the minimizer must belong to the subspace $\mathcal{H}_0$.

B.2 Proof of Theorem 1

The form of the solution in (3.19) is given by the Representer Theorem (see Appendix B.1. Plugging (3.19) into (3.18), the cost function amounts to

$$\begin{aligned}
&\lambda\|u\|_{\mathcal{H}_3^s}^2 + \sum_{j=1}^M \sum_{l\in\mathbb{I}^j} (\mathcal{T}_l^j u - q_l^j)^2 \\
=&\lambda\langle u, u\rangle + \sum_{j=1}^M \sum_{l\in\mathbb{I}^j} (\langle u, \phi_l^j\rangle - q_l^j)^2 \\
=&\lambda\langle \sum_{i=1}^M \sum_{k\in\mathbb{I}^i} c_k^i \phi_k^i, \sum_{j=1}^M \sum_{l\in\mathbb{I}^i} c_l^j \phi_l^j\rangle + \sum_{j=1}^M \sum_{l\in\mathbb{I}^j} (\langle \sum_{i=1}^M \sum_{k\in\mathbb{I}^i} c_k^i \phi_k^i, \phi_l^j\rangle - q_l^j)^2 \\
=&\lambda \sum_{i=1}^M \sum_{k\in\mathbb{I}^i} \sum_{j=1}^M \sum_{l\in\mathbb{I}^i} c_k^i c_l^j \langle \phi_k^i, \phi_l^j\rangle + \sum_{j=1}^M \sum_{l\in\mathbb{I}^j} (\sum_{i=1}^M \sum_{k\in\mathbb{I}^i} c_k^i \langle \phi_k^i, \phi_l^j\rangle - q_l^j)^2 \\
=&\lambda \mathbf{c}^\mathsf{T}\mathbf{\Phi}\mathbf{c} + \mathbf{c}^\mathsf{T}\mathbf{\Phi}^\mathsf{T}\mathbf{\Phi}\mathbf{c} - 2\mathbf{\Phi}\mathbf{q} + \mathbf{q}^\mathsf{T}\mathbf{q}.
\end{aligned} \tag{B.1}$$

By setting the gradient with respect to $\mathbf{c}$ to 0, we obtain

$$\mathbf{\Phi}\left(\mathbf{\Phi}+\lambda\mathbf{I}\right)\mathbf{c}=\mathbf{\Phi}\mathbf{q}. \tag{B.2}$$

The above holds since $\mathbf{\Phi}=\mathbf{\Phi}^{\mathsf{T}}$. (3.20) is certainly a solution to (B.2). □

B.3 Proof of Theorem 3

We present the proof of Theorem 3 in this section.

The form of the solution (4.18) is given by the Representer Theorem. Substituting (4.18) into equation (4.17), the coefficients c_k^i can be obtained by solving the constraint optimization problem

$$\begin{array}{ll}\text{minimize} & \frac{1}{2}\mathbf{c}^{\mathsf{T}}\mathbf{\Phi}\mathbf{c}\\ \text{subject to} & \mathbf{\Phi}\mathbf{c}=\mathbf{q}\end{array}. \tag{B.3}$$

We note that all solutions to $\mathbf{\Phi}\mathbf{c}=\mathbf{q}$ lead to the same value in $\frac{1}{2}\mathbf{c}^{\mathsf{T}}\mathbf{\Phi}\mathbf{c}$. Therefore, the solution to $\mathbf{\Phi}\mathbf{c}=\mathbf{q}$ verifies (B.3).

The necessary condition for perfect recovery can be more readily observed when we consider using the basis representation of $\mathbf{u}=[u_1(x,y,t),u_2(x,y,t),u_3(x,y,t)]^{\mathsf{T}}$ with

$$u_i(x,y,t)=\sum_{l_x=-L_x}^{L_x}\sum_{l_y=-L_y}^{L_y}\sum_{l_t=-L_t}^{L_t}d_{il_xl_yl_t}e_{l_xl_yl_t}(x,y,t), \tag{B.4}$$

Substituting (B.4) into equation (4.17), the coefficients $d_{il_xl_yl_t}$ in (B.4) have to verify the system of equations

$$\mathbf{\Xi}\mathbf{d}=\mathbf{q}, \tag{B.5}$$

where

$$\mathbf{d}=\begin{bmatrix}\mathbf{d}_1\\ \mathbf{d}_2\\ \mathbf{d}_3\end{bmatrix}$$

with $[\mathbf{d}_i]_l = d_{il}, i = 1, 2, 3$ and the column index l traverses all possible subscript combinations of l_x, l_y, l_t. $\boldsymbol{\Xi}$ is a block matrix

$$\boldsymbol{\Xi} = \begin{bmatrix} \boldsymbol{\Xi}^1 \\ \vdots \\ \boldsymbol{\Xi}^N \end{bmatrix},$$

and

$$\boldsymbol{\Xi}^i = \left[\boldsymbol{\Xi}_1^i, \boldsymbol{\Xi}_2^i, \boldsymbol{\Xi}_3^i\right],$$

with

$$\left[\boldsymbol{\Xi}_m^i\right]_{kl} = \langle \phi_{mk}^i, e_l \rangle \overset{(a)}{=} a_{mkl}^i,$$

for all $m = 1, 2, 3, i = 1, \cdots, M$, and the column index l traverses all possible subscript combinations of l_x, l_y, l_t, and (a) is given by (C.2). Since $\mathcal{H}_3^{s|3}$ is finite dimensional, the two approaches are equivalent in the absence of noise. This can be observed by noticing that $\boldsymbol{\Phi} = \boldsymbol{\Xi}\boldsymbol{\Xi}^{\mathsf{H}}$ (see also (C.7)).

The columns of $\boldsymbol{\Xi}$ are associated with the basis functions in $\mathcal{H}_3^{s|3}$ and the number of variables to be solved in this case is $dim(\mathcal{H}_3^{s|3})$. To achieve perfect reconstruction of any arbitrary $\mathbf{u}$, $\mathbf{d}$ must be uniquely determined and it is necessary to have $rank(\boldsymbol{\Xi}) = dim(\mathcal{H}^3)$. As each row of $\boldsymbol{\Xi}$ is essentially the sampling function $\boldsymbol{\phi}_k^i$, it is thereby necessary to have the set of sampling functions $\boldsymbol{\phi}$ span $\mathcal{H}_3^{s|3}$. Consequently, the number of rows must be greater than or equal to the number of columns, *i.e.*, the number of basis functions. Therefore, a necessary condition for perfect recovery is that the number of measurements/sampling functions must be at least $dim(\mathcal{H}_3^{s|3})$. This implies at least $dim(\mathcal{H}_3^{s|3}) + M$ spikes are needed with a neural circuit consists of M neurons. In addition, since each individual neuron encodes a temporal signal, at most $dim_t(\mathcal{H}_3^s)$ measurements per neuron are informative. Therefore, the number of neurons should at least be $dim(\mathcal{H}_3^{s|3})/dim_t(\mathcal{H}_3^s) = 3 \cdot dim_{xy}(\mathcal{H}_3^s)$. □

B.4 Proof of Equation (5.3)

Since $\mathcal{H}_3^s$ is an RKHS, by the reproducing property we have $u_m^i(x,y,t) = \langle u_m^i, \mathbf{K}_{xyt}\mathbf{e}_m\rangle_{\mathcal{H}_3^s}$. It follows that the mth term of the sum in Eq. (5.1) can be written as

$$\begin{aligned}
&\int_{\mathbb{D}} h_{1|m}(x,y,t-s)u_m^i(x,y,s)dsdxdy = \int_{\mathbb{D}} h_{1|m}(x,y,s)u_m^i(x,y,t-s)dxdyds \\
&\overset{(a)}{=} \int_{\mathbb{D}} h_{1|m}(x,y,s)u_m^i(x',y',t')K_m(x',y',t';x,y,t-s)dx'dy'dt'dxdyds \\
&\overset{(b)}{=} \int_{\mathbb{D}} u_m^i(x',y',t')\int_{\mathbb{D}} h_{1|m}(x,y,s)K_m(x,y,s;x',y',t-t')dxdydsdx'dy'dt' \\
&\overset{(c)}{=} \int_{\mathbb{D}} u_m^i(x',y',t')(\mathcal{P}_1^s\mathbf{h}_1)_m(x',y',t-t')dx'dy'dt' \\
&= \int_{\mathbb{D}} u_m^i(x',y',t-t')(\mathcal{P}_1^s\mathbf{h}_1)_m(x',y',t')dx'dy'dt',
\end{aligned}$$

where $^{(a)}$ follows from the reproducing property of the kernel K_m, $^{(b)}$ from the symmetry and the structure of the reproducing kernel $K_m(x',y',t';x,y,t)$ given in (3.4), and $^{(c)}$ from Definition 7.

B.5 Proof of Theorem 5

Proof: By the Riesz representation theorem [10], there exists a function $\phi_{1k}^i \in \mathcal{H}_1^s$ such that $\mathcal{T}_{1k}^i u_1 = \langle u_1, \phi_{1k}^i\rangle, \forall u_1 \in \mathcal{H}_1^s$. Moreover by the reproducing property

$$\phi_{1k}^i(t) = \langle \phi_{1k}^i, K_{1|t}^s\rangle = \mathcal{T}_{1k}^i\overline{K_{1|t}^s}.$$

Let $\mathcal{H}_{10}^s$ be a linear subspace of $\mathcal{H}_1^s$ spanned by ϕ_{1k}^i

$$\mathcal{H}_{10}^s = \text{span}\left(\{\phi_{1k}^i\}, k \in \mathbb{I}^i, i = 1,2\right)$$

and let $\mathcal{H}_{10}^{s\perp}$ be a linear subspace of $\mathcal{H}_1^s$ defined by

$$\mathcal{H}_{10}^{s\perp} = \{u_1 \in \mathcal{H}_1^s | \mathcal{T}_{1k}^i u_1 = 0, k \in \mathbb{I}^i, i = 1,2\}.$$

Then, for any $u_1 \in \mathcal{H}_{10}^{s\perp}$ and any $\sum_{i=1}^2\sum_{k\in\mathbb{I}^i} c_k^i\phi_{1k}^i \in \mathcal{H}_{10}^s$, we have

$$\langle u_1, \sum_{i=1}^2\sum_{k\in\mathbb{I}^i} c_k^i\phi_{1k}^i\rangle = \sum_{i=1}^2\sum_{k\in\mathbb{I}^i} c_k^i\langle u_1, \phi_{1k}^i\rangle = \sum_{i=1}^2\sum_{k\in\mathbb{I}^i} c_k^i\mathcal{T}_{1k}^i u_1 = 0.$$

Since $\mathcal{H}_1^s = \mathcal{H}_{10}^s \oplus \mathcal{H}_{10}^{s\perp}$, u_1 can be represented as $u_1 = u_{10} + u_{10}^\perp$ where $u_{10} \in \mathcal{H}_{10}^s$ and $u_{10}^\perp \in \mathcal{H}_{10}^{s\perp}$ are orthogonal. Therefore,

$$\|u_{10} + u_{10}^\perp\|^2 = \|u_{10}\|^2 + \|u_{10}^\perp\|^2.$$

Similarly, there exists a function $\phi_{2k}^i \in \mathcal{H}_2^s$ such that $\mathcal{T}_{2k}^i u_2 = \langle u_2, \phi_{2k}^i \rangle$, where $\phi_{2k}^i(t_1, t_2) = \mathcal{T}_{2k}^i \overline{K_{2|t_1 t_2}^s}$. u_2 can be represented as $u_2 = u_{20} + u_{20}^\perp$, where $u_{20} \in \mathcal{H}_{20}^s$ and $u_{20}^\perp \in \mathcal{H}_{20}^{s\perp}$ are orthogonal, with

$$\mathcal{H}_{20}^s = \text{span}(\{\phi_{2k}^i\}, k \in \mathbb{I}^i, i = 1, 2),$$

and

$$\mathcal{H}_{20}^{s\perp} = \{u_2 \in \mathcal{H} | \mathcal{T}_{2k}^i u_2 = 0, k \in \mathbb{I}^i, i = 1, 2\}.$$

Finally,

$$\begin{aligned}
&\sum_{i=1}^{2} \sum_{k \in \mathbb{I}^i} \left(\mathcal{T}_{1k}^i u_{10} + \mathcal{T}_{2k}^i u_{20} - q_k^i\right)^2 + \lambda_1 \|u_{10}\|_{\mathcal{H}_1^s}^2 + \lambda_2 \|u_{20}\|_{\mathcal{H}_2^s}^2 \\
=&\sum_{i=1}^{2} \sum_{k \in \mathbb{I}^i} \left(\mathcal{T}_{1k}^i (u_{10} + u_{10}^\perp) + \mathcal{T}_{2k}^i (u_{20} + u_{20}^\perp) - q_k^i\right)^2 + \lambda_1 \|u_{10}\|_{\mathcal{H}_1^s}^2 + \lambda_2 \|u_{20}\|_{\mathcal{H}_2^s}^2 \\
\leq&\sum_{i=1}^{2} \sum_{k \in \mathbb{I}^i} \left(\mathcal{T}_{1k}^i (u_{10} + u_{10}^\perp) + \mathcal{T}_{2k}^i (u_{20} + u_{20}^\perp) - q_k^i\right)^2 + \lambda_1 \|u_{10} + u_{10}^\perp\|_{\mathcal{H}_1^s}^2 + \lambda_2 \|u_{20} + u_{20}^\perp\|_{\mathcal{H}_2^s}^2
\end{aligned}$$

Therefore, the minimizer to (6.28) must belong to the subspaces $\mathcal{H}_{10}^s$ and $\mathcal{H}_{20}^s$.

By plugging (3.19) into (6.28) and setting the gradient with respect to $\mathbf{c}$ to 0, we see that $\mathbf{c}$ is the solution to (6.30). □

B.6 Proof of Theorem 9

We present the proof of Theorem 9 in this section.

Since the filterbank is structured, we will use the compact notation in Section 8.1.2.3 as (k, m, n, l) with $k \in \mathbb{N}, m \in \mathbb{Z}, n \in \mathbb{Z}/N, l \in \mathbb{Z}/L$. Moreover, since the discretized subset is countable, the elements of the filterbank can be ordered in one dimension with an injective mapping $y : \mathbb{N} \times \mathbb{Z} \times \mathbb{Z}/N \times \mathbb{Z}/L \mapsto \mathbb{Z}$, where $\mathbb{Z}/N$ and $\mathbb{Z}/L$ denote the set of integers modulo

N and L, respectively. The above transformations correspond to a connectivity setting of the switching matrix such that

$$y^{-1}(\sigma(y(k,m,n,l))) - (k,m,n,l) = const. \tag{B.6}$$

The interpretation of (B.6) is that for any operator $\mathcal{T}$ described above, i.e., $\mathcal{T} \in H_p$, acting on the input video u, it is sufficient to channel the spikes coming from the neural circuit j to the l-th entry of the next level, i.e., $\sigma(i) = l$, where l satisfies $h_1^i = \mathcal{T} h_1^l$. Since the set of spatial receptive fields is invariant under any $\mathcal{T} \in H_p$, such an l always exists and is given by (8.13) and (B.6) depending on the transformation. Moreover each l is unique, i.e., the permutation σ is injective and σ^{-1} exists.

It remains to prove that the set of "switched" spike trains

$$\sigma(\mathbb{T}) = \left\{ (t_k^{\sigma^{-1}(i)}), k \in \mathbb{Z}, i = 1, 2, \ldots, M \right\}$$

represents the video input $\mathcal{T}u$. We see that $v^i(t)$ which produces the spike train (t_k^i) can be expressed as

$$v^j(t) = \int_{\mathbb{R}} \left(\int_{\mathbb{R}^2} h_1^j(x,y,s) u(x,y,t-s)\, dxdy \right) ds. \tag{B.7}$$

Similarly to (B.7) let us define $v_{\mathcal{T}}^i(t)$ as

$$v_{\mathcal{T}}^i(t) = \int_{\mathbb{R}} \left(\int_{\mathbb{R}^2} {}^s h_1^i(x,y) \cdot {}^\tau h_1^i(s) (\mathcal{T}u)(x,y,t-s)\, dxdy \right) ds, \tag{B.8}$$

i.e., the output of the i-th receptive field when the input is $\mathcal{T}u$ (here we also used the space-time separability of the receptive fields). Since $\mathcal{T}$ is a unitary operator we have that

$$v_{\mathcal{T}}^i(t) = \int_{\mathbb{R}} \left(\int_{\mathbb{R}^2} (\mathcal{T}^{-1}({}^s h_1^i(x,y))) \cdot {}^\tau h_1^i(s) u(x,y,t-s)\, dxdy \right) ds = v^{\sigma^{-1}(i)}(t). \tag{B.9}$$

In other words, the input $\mathcal{T}u$ at the i-th spiking circuit produces the spike train $(t_k^{\sigma^{-1}(i)})$ and therefore the set $\sigma(\mathbb{T})$ represents the video input $\mathcal{T}u$. □

Appendix C

Algorithmic Constructions

C.1 Computation of the Sampling Functions and Φ matrix

To compute the entries for matrix $\mathbf{\Phi}$ in (4.19), we note from (4.16) that $\phi^i_{mk}(x_1, x_2, t)$ amounts to

$$
\begin{aligned}
&= \int_{t^i_k}^{t^i_{k+1}} \left(\int_{\mathbb{D}} h^i_{1|m}(x', y', s-t') K_m(x, y, t; x', y', t') dx' dy' dt' \right) ds \\
&= \int_{t^i_k}^{t^i_{k+1}} \left(\int_{\mathbb{D}} h^i_{1|m}(x', y', t') K_m(x, y, t; x', y', s-t') dx' dy' dt' \right) ds \\
&= \sum_{l_x=-L_x}^{L_x} \sum_{l_y=-L_y}^{L_x} \sum_{l_t=-L_t}^{L_t} \int_{t^i_k}^{t^i_{k+1}} \left(\int_{\mathbb{D}} h^i_{1|m}(x', y', t') e_{l_x l_y l_t}(x-x', x-y', t+t'-s) dx' dy' dt' \right) ds \\
&= \sum_{l_x=-L_x}^{L_x} \sum_{l_y=-L_y}^{L_y} \sum_{l_t=-L_t}^{L_t} e_{l_1 l_2 l_t}(x, y, t) \int_{t^i_k}^{t^i_{k+1}} \left(\int_{\mathbb{D}} h^i_{1|m}(x', y', t') e_{-l_x, -l_y, -l_t}(x', y', s-t') dx' dy' dt' \right) ds \\
&= \sum_{l_x=-L_x}^{L_x} \sum_{l_y=-L_y}^{L_y} \sum_{l_t=-L_t}^{L_t} e_{l_x l_y l_t}(x, y, t) \int_{t^i_k}^{t^i_{k+1}} e_{-l_t}(s) ds \int_{\mathbb{D}} h^i_{1|m}(x', y', t') e_{-l_x, -l_y, l_t}(x', y', t') dx' dy' dt'.
\end{aligned}
\tag{C.1}
$$

Since the $e_{l_x l_x l_t}(x, y, t)$'s form the orthonormal base in $\mathcal{H}$, we see that

$$
\phi^i_{mk}(x, y, t) = \sum_{l_x=-L_x}^{L_x} \sum_{l_y=-L_y}^{L_y} \sum_{l_t=-L_t}^{L_t} a^i_{mkl_x l_y l_t} e_{l_x, l_y, l_t}(x, y, t), \tag{C.2}
$$

where $a^i_{mkl_x l_y l_t}$ are the coefficients of the linear combination of bases and

$$
a^i_{mkl_x l_y l_t} = \int_{t^j_k}^{t^j_{k+1}} e_{-l_t}(s) ds \cdot \int_{\mathbb{D}} h^i_{1|m}(x', y', t') e_{-l_x, -l_y, l_t}(x', y', t') dx' dy' dt', \tag{C.3}
$$

or

$$a^i_{mkl_xl_y,-l_t} = \left(\int_{t^i_k}^{t^i_{k+1}} e_{l_t}(s)ds\right)\left(\int_{\mathbb{D}} h^i_{1|m}(x',y',t')e_{-l_x,-l_y,-l_t}(x',y',t')dx'dy'dt'\right). \quad \text{(C.4)}$$

Let

$$h^i_{ml_xl_yl_t} = \int_{\mathbb{D}} h^i_{1|m}(x',y',t')e_{-l_x,-l_y,-l_t}(x',y',t')dx'dy'dt', \quad \text{(C.5)}$$

we have

$$a^i_{mkl_xl_yl_t} = \begin{cases} \left(t^i_{k+1}-t^i_k\right)h^i_{ml_xl_y,-l_t}, & l_t = 0 \\ \dfrac{jL_t}{\Omega_t l_t}\left(e_{-l_t}(t_{k+1}) - e_{-l_t}(t_k)\right)h^i_{ml_xl_y,-l_t}, & l_t \neq 0 \end{cases} \quad \text{(C.6)}$$

The computation of the coefficients in (C.5) can be simplified by considering the space-time domain $\mathbb{D}$ to be exactly one period of the function in $\mathcal{H}^s_3$, and by numerically evaluating the integral in the second half of (C.4) using the rectangular rule with uniform grid. Since the result is closely related to the 3D-DFT coefficients of $h^i_{1|m}(x,y,t)$, these coefficients can be very efficiently obtained. Note also that the $a^i_{mkl_xl_yl_t}$ clearly depends on the particular neuron model and the spatio-temporal receptive field used in the encoding. Equation (C.4) shows, however, that this dependency can easily be separated into two terms. The term in the first parenthesis depends only on the IAF neuron and the term in the second parenthesis depends only on the receptive field.

Therefore,

$$\begin{aligned} \left[\boldsymbol{\Phi}^{ij}\right]_{kl} &= \langle \boldsymbol{\phi}^i_k, \boldsymbol{\phi}^j_l \rangle_{\mathcal{H}^{s|3}_3} \\ &= \sum_{m=1}^{3} \langle \phi^i_{mk}, \phi^j_{ml} \rangle_{\mathcal{H}^s_3} \\ &= \sum_{m=1}^{3} \sum_{l_x=-L_x}^{L_x} \sum_{l_y=-L_y}^{L_y} \sum_{l_t=-L_t}^{L_t} a^i_{mkl_xl_yl_t} \overline{a^j_{mll_xl_yl_t}}. \end{aligned} \quad \text{(C.7)}$$

C.2 Constructing Gabor Receptive Fields and IAF Neurons

Each receptive field component has a profile modeled as a spatial Gabor filter derived from the mother function

$$h(x,y,\eta) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{x^2}{2}-\frac{y^2}{8}\right)\cos(-2.5x+\eta),$$

with translations

$$\tau_{(x_0,y_0)}h(x,y,\eta) = h(x-x_0, y-y_0, \eta),$$

dilations

$$\mathcal{D}_\alpha h(x,y,\eta) = \frac{1}{\alpha}h\left(\frac{1}{\alpha}x, \frac{1}{\alpha}y, \eta\right),$$

and rotations

$$\mathcal{R}_\theta h(x,y,\eta) = h\left(\cos(\theta)x + \sin(\theta)y, -\sin(\theta)x + \cos(\theta)y, \eta\right).$$

The phase η provides additional flexibility in modeling phase selectivity of Gabor receptive fields [159].

We consider an initial orientation θ_m^i and phase η_m^i picked from a uniform distribution $[0, 2\pi)$, as well as two levels of dilation $\alpha_m^i \in \{2^{0.5}, 2^{1.5}\}$, with probability 0.8 and 0.2, respectively. The center coordinates of the red-component receptive fields (x_0^i, y_0^i) are picked randomly from a uniform distribution. The center coordinates of green- and blue-component receptive fields are picked around the red-component center with Gaussian distributions $\mathcal{N}(\mathbf{0}, \mathbf{I})$, where $\mathbf{I}$ is 2×2 identity matrix. Note that while the parameters above are randomly chosen, once picked they are assumed to be known (or identifiable).

To create a non-separable spatio-temporal receptive field, we add a temporal component to the Gabor functions such that the receptive fields rotate at an angular speed $v = 2.5\pi(rad/s)$ around their respective centers (x_m^i, y_m^i). Furthermore, the temporal dynamics is modulated by a raised cosine function

$$f(t) = \begin{cases} 1 - \cos(2\pi \cdot 10 \cdot t), & 0 \le t \le 0.1[s] \\ 0, & otherwise \end{cases}$$

to ensure that the spatiotemporal receptive field is causal in the time variable and has finite memory.

The overall receptive field can be expressed as

$$h_{1|m}^i(x,y,t) = f(t)\tau_{x_m^i, y_m^i}\mathcal{D}_{\alpha_m^i}\mathcal{R}_{\theta_m^i + 2.5\pi t}h(x,y,\eta_m^i).$$

The bias, threshold and integration constant of all IAF neurons are picked to be the same, and they are $b^i = 3$, $\delta^i = 0.1$, and $\kappa^i = 1$, respectively. In simulations, since the

input video had a frame rate of 100 frames per second, the inputs to the IAF neurons had a time step of 0.01 second. The time occurrences of spikes generated by the IAF neurons are analytically computed using linear interpolation between two consecutive time steps. The encoding process took 5 minutes on a single Nvidia M2050 GPU.

C.3 Segmenting Continuous Visual Stimuli for Identification

We use a sliding temporal window to create multiple video clips from a single continuous natural video. This is needed to fix one of the complications arising in using natural video with the introduced methodology, namely how to properly segment a long natural sequence into multiple segments of videos. Since the spatio-temporal receptive field has temporal memory of length $S \triangleq \text{supp}(\mathbf{h})$, i.e., it extends S seconds into the past, the timing of a spike at a time t_k is affected by the stimulus on the time interval of length S preceding the spike, i.e., by values of the stimulus $\mathbf{u}(t)$ on $t \in (t_k - S, t_k]$. Therefore, when recording spikes in response to a stimulus $\mathbf{u}(t)$, care should be taken so that the recording is longer than the temporal support of the receptive field and only those spikes occurring S seconds after the start of the recording are used.

In Figure C.1(a), we illustrate the valid spikes given a window size S_t and filter size S: The sliding window is of the same length as the temporal period of the RKHS and is shown in blue. The corresponding clip of the signal $u(t)$ is highlighted in blue. The filter $h(t)$ (the shape of $h(t)$ in Figure C.1(a) is shown for illustration purposes only) has temporal support of S. From all spikes generated in $[0, S_t^s]$, only the spikes generated in the interval $(S, S_t^s]$ (green spikes) can be used in identification. Phrased differently, if the window size is S_t and one uses spikes generated in the interval $(R, S_t^s]$ in identification, then the identified receptive field is only valid if the temporal support of it is within $[0, R]$.

The sliding window size we choose is $0.2s$ and the step between windows is $0.1s$, as schematically shown in Figure C.1(b), where the color of the spikes indicates its use in the corresponding window. Note that practically no spikes are discarded as the windows overlap.

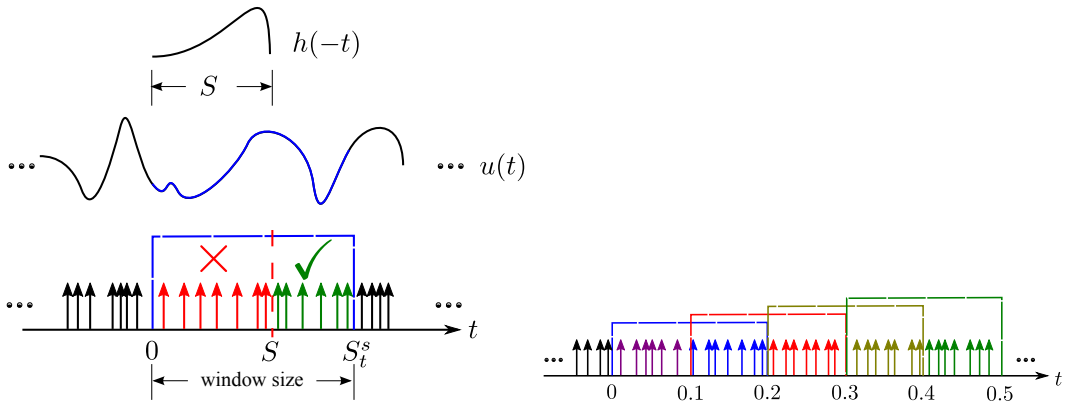


Figure C.1: Schematic illustration of segmenting a continuous visual stimulus used in identification into overlapping video clips and choosing the valid spikes for each video clip. (a) If the temporal support of the receptive field is S (top), one can choose a video clip of duration $S_t^s > S$. The spikes that are valid for this video clip are shown in green at the bottom. Although the spikes indicated in red are generated during the presentation of this video clip, they are not valid since they contain information outside the duration of this video clip. (b) In identification, a continuous stream of visual stimuli can be presented to the neural circuit. To perform identification, the continuous visual stimulus needs to be segmented into overlapping video clips with appropriate spikes chosen for each of the video clips. Here, an example with window size 0.2 [s] and $S = 0.1$ [s] is given. The cut-off times of each video clip are shown with rectangular boxes of various colors. The valid spikes for each video clip are shown in their respective color. As a result, only a small number of measurements are discarded.

Appendix D

Additional Simulation Results

D.1 Example of Identification of Feedforward and Feedback DSPs under Noiseless Condition

We provide here an example of functional identification of the neuron circuit 1 in example 3 under noiseless condition; the same DSP kernels and Hodgkin-Huxley neuron model were used.

First, we use the circuit to encode a 50 [Hz] input signals of duration 0.4 [s]. We repeat this for $1,000$ times with a different, randomly generated input each time.

In the $1,000$ trials, a total of $18,964$ spikes are generated by BSG 1 and $25,271$ spikes are generated by BSG 2. We deem the maximum valid interspike interval to be 13.713[msec]. Among all the spikes, valid measurements amount to $9,615$ for BSG 1 and $8,093$ for BSG 2. The identified DSP kernels are shown in Figure D.1. We also show identification quality against the number of valid measurements in Figure D.1(i). The identification quality saturates after using more than $2,000$ measurements. This correspond to about 200 trials.

The identification results for the DSP kernels associated with Neuron 2 are shown in Figure D.2.

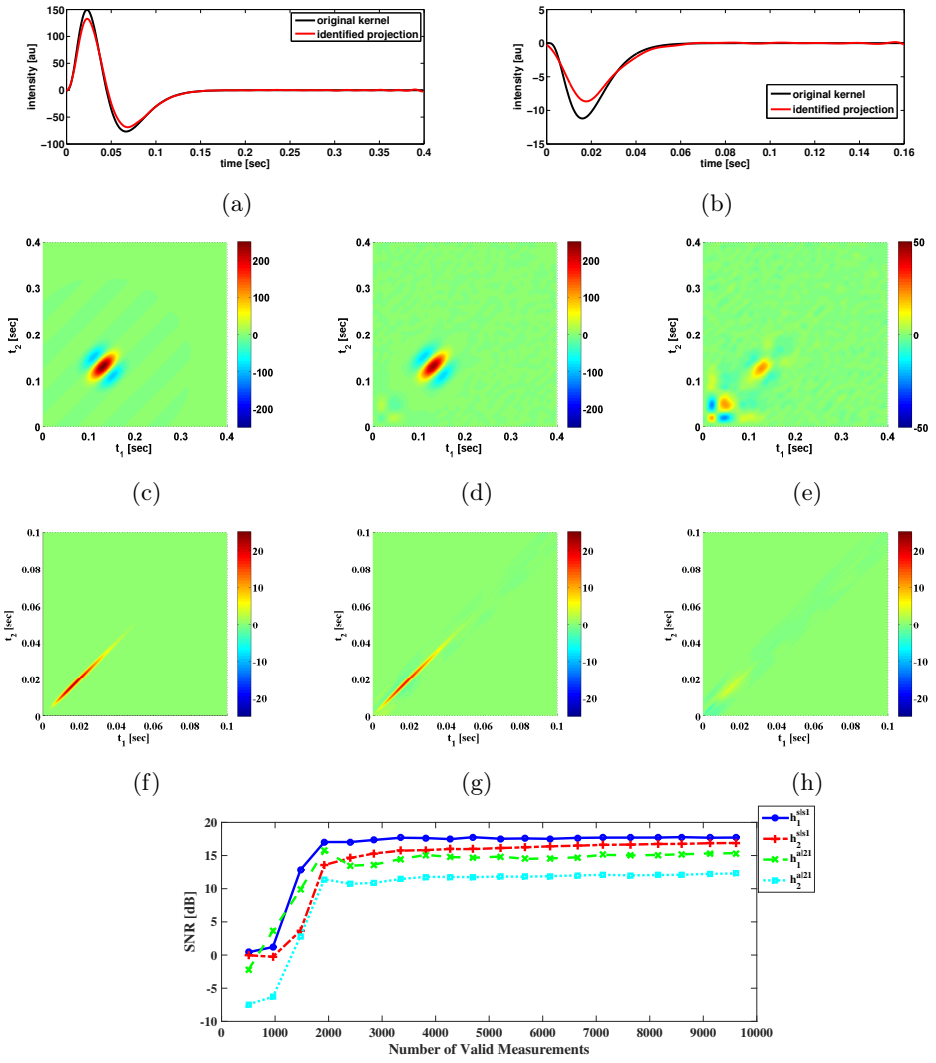


Figure D.1: Examples of functional identification of a neural circuit with Hodgkin-Huxley neurons as spike generators under noiseless condition (Neuron 1). (a) Original first order feedforward kernel (black) and identified projection of the kernel (red). (b) Original first order feedback kernel (black) and identified projection of the kernel (red). (c) Original second order feedforward kernel. (d) Identified projection of second order feedforward kernel. (e) Error of identified second order feedforward kernel. (f) Original second order feedback kernel. (g) Identified projection of second order feedback kernel. (h) Error of identified second order feedback kernel. (i) SNR of the identified DSP kernels that feed into Neuron 1 against number of valid spikes used in identification.

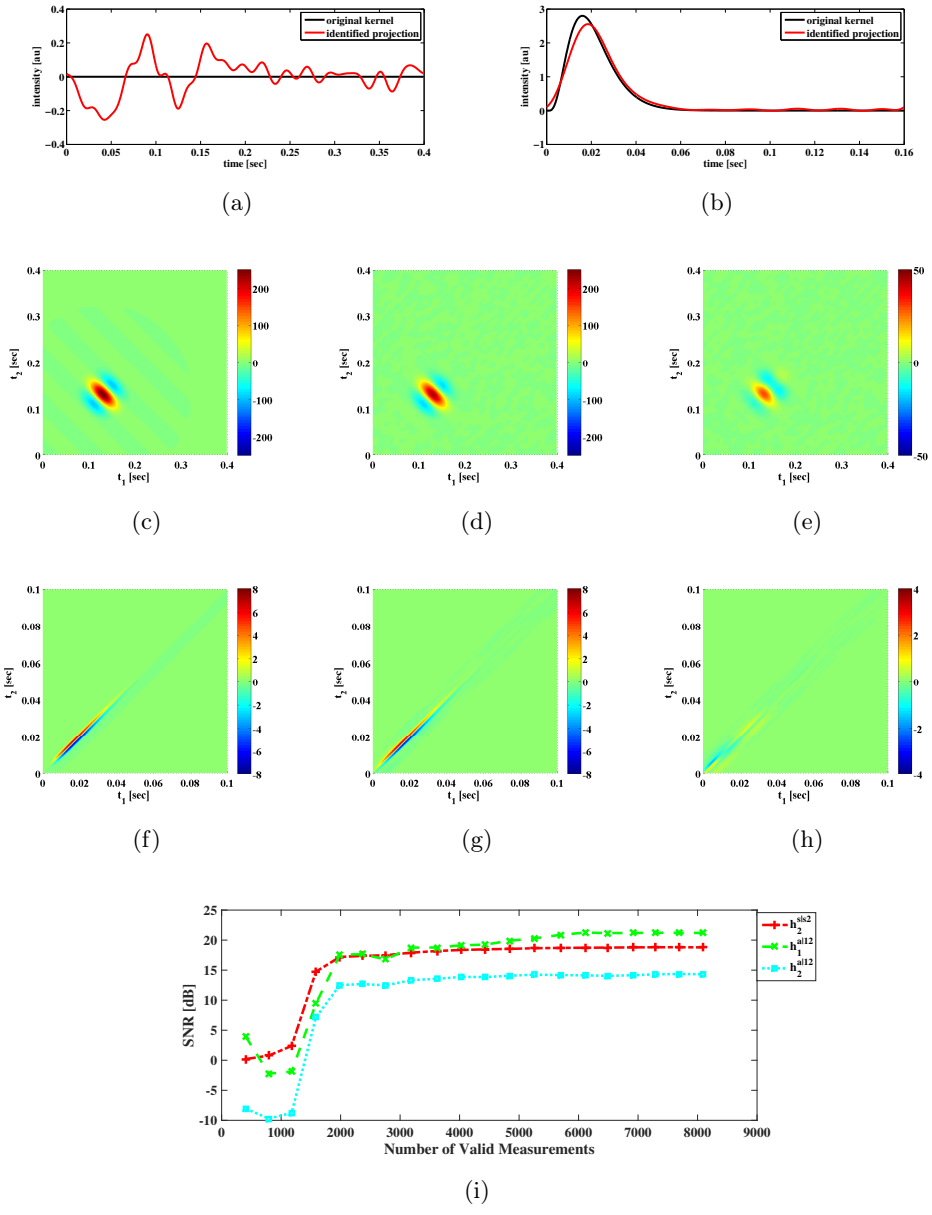


Figure D.2: Examples of functional identification of a neural circuit with Hodgkin-Huxley neurons as spike generators under noiseless condition (Neuron 2). (a) Original first order feedforward kernel (black) and identified projection of the kernel (red). Note that the original first order kernel is zero. Compare scale of y-axis to that in Figure D.1(a).

Figure D.2: (Continued) (b) Original first order feedback kernel (black) and identified projection of the kernel (red). (c) Original second order feedforward kernel. (d) Identified projection of second order feedforward kernel. (e) Error of identified second order feedforward kernel. (f) Original second order feedback kernel. (g) Identified projection of second order feedback kernel. (h) Error of identified second order feedback kernel. (i) SNR of the identified DSP kernels that feed into Neuron 2 against number of valid spikes used in identification. $h_1^{s|s2}$ is omitted since it is zero.

D.2 Example of Identification of Feedforward and Feedback DSPs With Integrate-and-Fire Neuron

We also provide here an example using Integrate-and-Fire (IAF) Neurons instead of using Hodgkin-Huxley neurons as in Section D.1. Since the t-transforms of encoding with IAF neurons are exact, we show here that the identification algorithm leads to perfect identification when appropriate input stimuli and spike spaces are used.

Figure D.3 shows the identification result for neuron 1 using the same input stimulus spaces and spike spaces as in Section D.1.

Figure D.4 shows the identification result for neuron 1 using a higher bandwidth for the spike space when compared with the one used in Figure D.3.

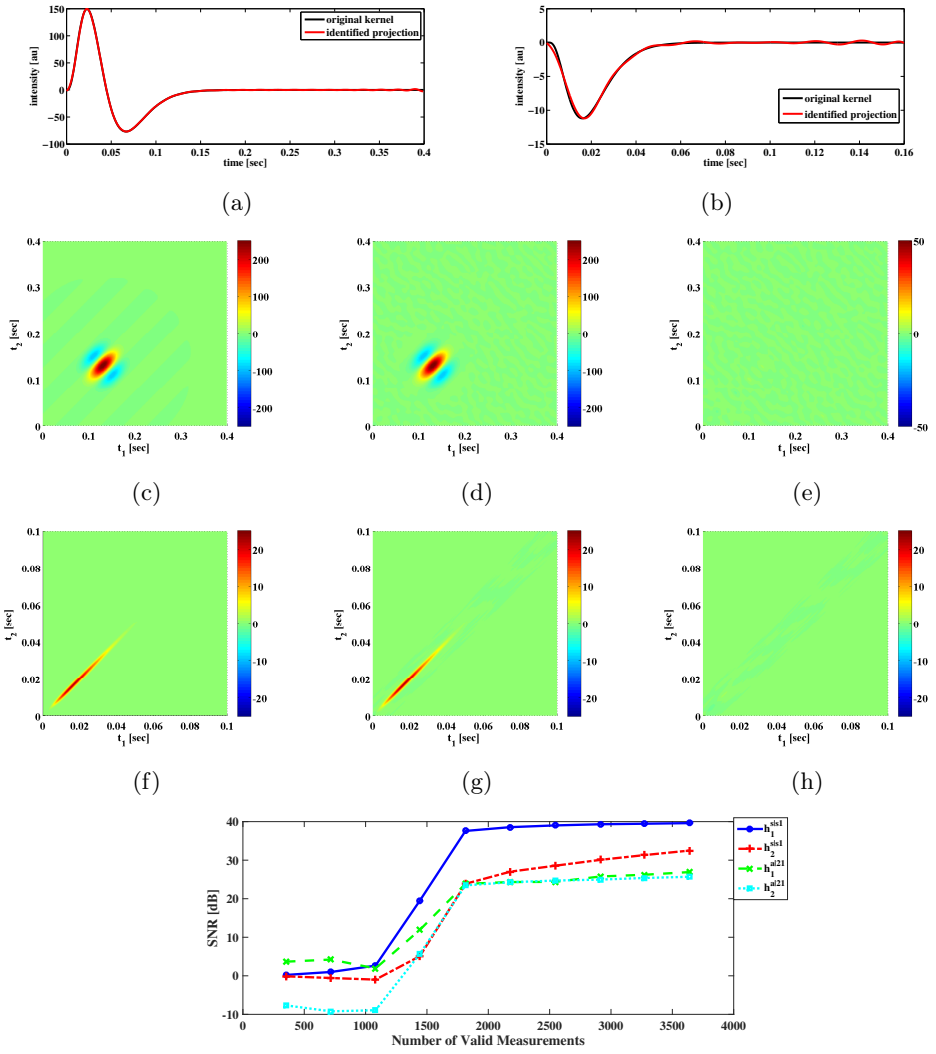


Figure D.3: Examples of functional identification of a neural circuit with IAF neurons as spike generators under noiseless condition (Neuron 1). The IAF neurons has a refractory period of 2 [msec]. (a) Original first order feedforward kernel (black) and identified projection of the kernel (red). (b) Original first order feedback kernel (black) and identified projection of the kernel (red). (c) Original second order feedforward kernel. (d) Identified projection of second order feedforward kernel. (e) Error of identified second order feedforward kernel. (f) Original second order feedback kernel. (g) Identified projection of second order feedback kernel. (h) Error of identified second order feedback kernel. (i) SNR of the identified DSP kernels that feed into Neuron 1 against number of valid spikes used in identification.

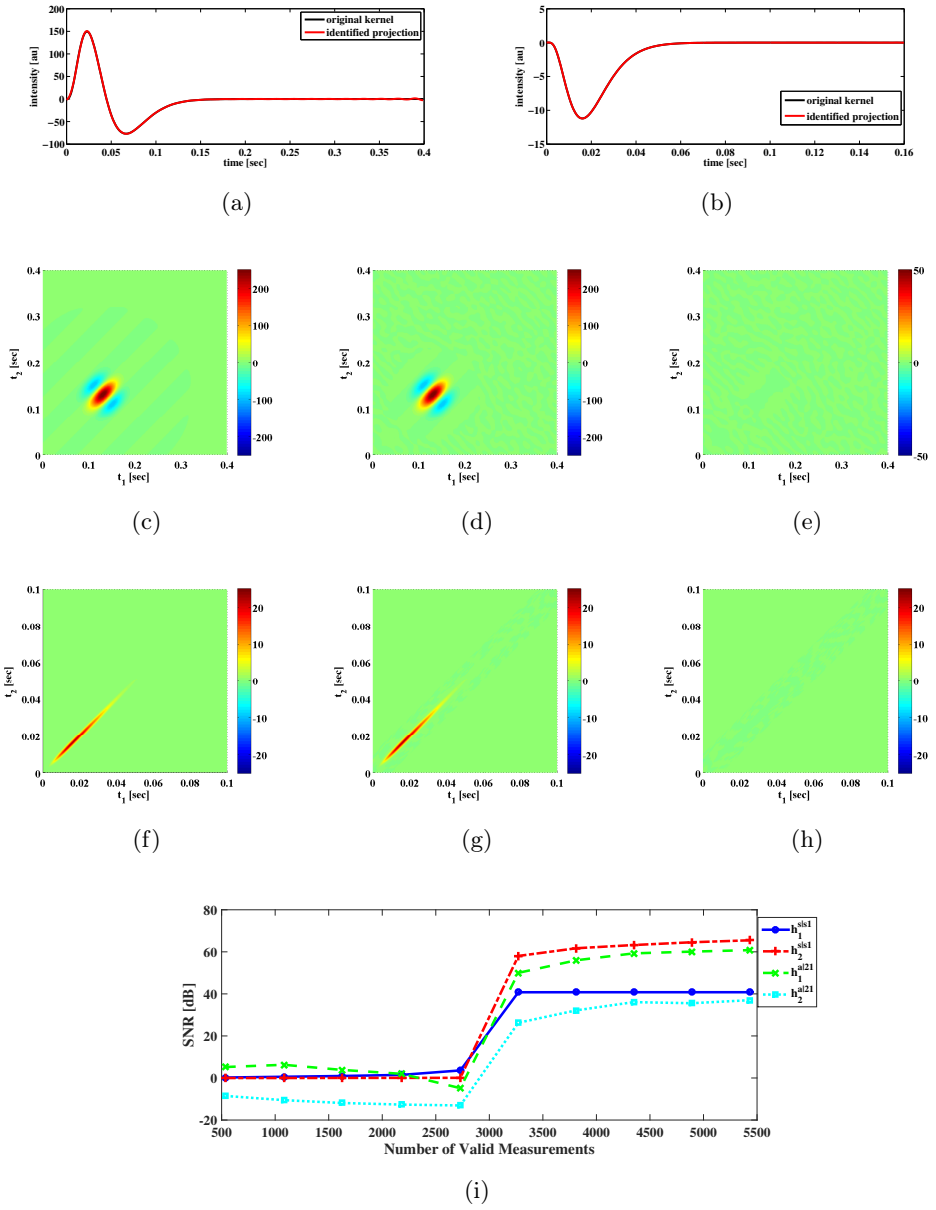


Figure D.4: Examples of functional identification of a neural circuit with IAF neurons as spike generators under noiseless condition (Neuron 1). The IAF neurons has a refractory period of 2 [msec]. Compared to the example in Figure D.3, we increased the bandwidth of the space $\mathcal{H}_1^{a|21}$, *i.e.*, the space of spikes, to better approximate the effect of feedback spikes on the dendritic current. One can see that the quality of the identified feedback kernels further increases. So does the quality of the identified feedforward kernels. The sum of the orders of the kernel spaces is 2982.

Figure D.4: (Continued) (a) Original first order feedforward kernel (black) and identified projection of the kernel (red). (b) Original first order feedback kernel (black) and identified projection of the kernel (red). (c) Original second order feedforward kernel. (d) Identified projection of second order feedforward kernel. (e) Error of identified second order feedforward kernel. (f) Original second order feedback kernel. (g) Identified projection of second order feedback kernel. (h) Error of identified second order feedback kernel. (i) SNR of the identified DSP kernels that feed into Neuron 1 against number of valid spikes used in identification.