Online characterization of the retinal network

Baptiste Lefebvre

PSL
UNIVERSITÉ PARIS

THÈSE DE DOCTORAT
DE L'UNIVERSITÉ PSL

Préparée au Laboratoire de physique statistique de l'ENS
et à l'Institut de la vision

Caractérisation en ligne du réseau rétinien

Online characterization of the retinal network

Soutenue par
Baptiste LEFEBVRE
le **11 décembre 2020**

École doctorale n°158
Cerveau, cognition, comportement

Spécialité
Neurosciences

Composition du jury

Alexandre GRAMFORT Président
Directeur de recherche, INRIA Saclay

Tim GOLLISCH Rapporteur
Professor, University Medical Center Göttingen

Felix FRANKE Rapporteur
Assistant Professor, IOB / University of Basel

Wei WEI Examinatrice
Associate Professor, University of Chicago

Thierry MORA Directeur de thèse
Directeur de recherche, Laboratoire de physique statistique de l'ENS

Olivier MARRE Directeur de thèse
Directeur de recherche, Institut de la vision

Pierre YGER Invité
Chargé de recherche, Institut de la vision

ENS | PSL

To my family

Contents

List of Figures

Introduction

A major area in computational neuroscience aims at understanding the information content of neural signals. From sensory to motor parts of the central nervous system, these signals come in a standard form: a sequence of electrical pulses called action potentials or spikes. To understand their meaning, we need to understand the structure of spike patterns and unravel the rules that would define what could be called the neural syntax. This approach should reveal how we receive, process, store and transmit information from the external world.

In the last several decades, advances in experimental techniques opened new horizons for studying the neural code. Our ability to record the activity of large populations of neurons has greatly improved. This is particularly true in electrophysiology where silicon probes and microelectrode arrays now provide a high number of densely packed recording sites. As a result, the volume of recorded neural data increased enormously and challenged traditional data analysis processes.

A particularly suitable system to study how sensory neurons encode visual information is the retina. This neural system transforms the light patterns entering the eye into sequences of spikes which are sent to the brain. Visual information is only carried by this sensory system with almost no feedback from the brain, which allows us to study its visual processing in isolation. Moreover the flatness of the retinal tissue makes it perfectly suitable to record its activity with microelectrode arrays. The responses recorded from large populations of ganglion cells, the output of the retina, are noisy and show a high degree of complexity. In fact, around 30 different cell types have been identified and each of them is supposed to encode a different feature of the visual scene. These representations are often nonlinear and adapt to visual statistics such as the mean luminance. These properties make the retinal code hard to decipher and a matter of active research.

In this thesis work, I first review the signal processing algorithms that have been proposed to reconstruct the electrical activity of individual neurons from extracellular voltage traces recorded with high-density electrophysiolog-

ical devices. This inverse problem of source separation is referred to as spike sorting. I identify the common properties and the main differences to finally outline the issues that remain to be solved. In a following part, I present a generic toolbox for online spike sorting based on a combination of density clustering, greedy template matching and parallel computing. This allows us to resolve the spike trains of each recorded cells in firm real-time.

In a last part, I present a specific application on the retina where online spike sorting might be useful. Classically, ganglion cells are supposed to extract specific features from the visual scene such as increases or decreases of luminance, ON or OFF cells respectively. However, retinal processing depends on the visual context. Using a novel perturbative approach, I show that the same cell can turn ON or OFF depending on the natural context. I found that a convolutional neural network model fitted to the data can recapitulate this context dependence. Online perturbations are thus a promising tool to probe computations in the retina but also more widely in other sensory systems.

This thesis is organized as follows:

Chapter 1 I begin with an introduction of the physiology of the retina. I show that complex functions emerge from the cell type diversity and the network structure of this sensory system.

Chapter 2 I review algorithms used to recover the activity of individual neurons from extracellular voltage traces recorded with dense and large-scale microelectrode arrays.

Chapter 3 I present a generic toolbox for online spike sorting which allows resolving the spike trains of each recorded cells in firm real-time.

Chapter 4 I study the stability of feature extraction by retinal ganglion cells in different natural contexts with fixed mean luminance and contrast. In particular, I show that some cells can respond selectively to small increases of light in some contexts and to decreases in others.

Acknowledgments

Even if this thesis manuscript has a single author, the research presented here would not have been possible without the direct or indirect support of many people. I wouldn't be the person I am today and I wouldn't be where I am now without them.

First, I would like to warmly thank my thesis supervisors, Olivier Marre and Thierry Mora, for their mentorship during all those years. You taught me and you supported me a lot in my discovery of both experimental and theoretical neuroscience. I am particularly grateful for the examples of scientific enthusiasm, intellectual rigor and kindness you gave me. Your advices will follow me in my life after the PhD.

I would also like to particularly thank Pierre Yger, my unofficial supervisor, who introduced me to the joy of spike sorting. Thanks for your communicative passion and your unwavering enthusiasm. There's no doubt that I will miss our daily discussions.

I would like to thank the members of Olivier's team for all the vivid discussions we had and the life you brought to the lab: Christophe Gardella, Stéphane Deny, Giulia Spampinato, Ulisse Ferrari, Matthew Chalk, Elaine Orendorff, Tristan Fauvel, Danica Despotović, Francesco Trapani, Semihcan Sermet, Thomas Buffet, Déborah Varro, Gariel Mahuas, Mathieu Pham Van Cang and Matías Goldin. You have all been cheerful and talented colleagues.

I would also like to thank Romain Brette for hosting and including me in his lab. I've learned a lot about computational neuroscience from you and all the members of your team: Pierre Yger, Marcel Stimberg, Maria Telenczuk, Charlotte Le Mouel, Sarah Goethals, Anirudh Kulkarni, Nissim Zerbib, Huy-Hoang Nguyen and Irene Elices. Thank you for the pleasant work atmosphere you provide me and all the moments we shared together both inside and outside the lab.

I would like to thank additional people from the Institut de la Vision. Romain Caplette, Corentin Joffrois, Valérie Fradot and Quenol Cesar for their excellent technical support in the experiments. Brigitte Levy, Sylvain Duval, Nadia Nahnouh and Nathalie Boyer for their excellent administrative support. Kim Nguyen-Ba-Charvet for inviting me to participate to the Brain Awarness Week.

I would like to thank Yannick Andéol for his in-depth knowledge of axolotls and his peer support.

I would like to thank the neuroscientists I had the occasion to meet in Göttingen, in particular Fred Wolfe and Alexander Ecker.

I would like to thank Nicolas Maudet and Antoine Miné who gave me the opportunity to experience teaching at Sorbonne University.

I would like to thank Paris Science et Lettres (PSL) University and the Fondation pour la Recherche Médicale (FRM) for their financial support.

I would like to thank Valérie Ego-Stengel and Alexandre Gramfort for their guidance during my yearly thesis committees.

I would like to thank Tim Gollisch, Felix Franke, Wei Wei and Alexandre Gramfort, who accepted to be members of my jury and to take the time to evaluate my work.

Finally I would like to thank all my friends and my large family for their encouragements. I am extremely grateful to my parents and my sisters Alice, Amandine and Julie for their active support in my scientific journey, and Elodie for sharing my life during those years.

Chapter 1

Physiology of the retina

The eyes are our windows on the world. All the visual information we need to interpret and to interact with our surrounding environment is received by these sensory organs. Light patterns enter the eye through the cornea and lens. They provide the refractive optics that project images onto the retina lining the back of the eyeball. This light-sensitive neural tissue detects and processes the incoming light into neural signals which are then sent to the brain through the optic nerve (figure 1.1). As such, the retina is a bottleneck: any visual information it loses can never be recovered further downstream.

1.1 Anatomy

1.1.1 Layers and cell types

The retina is a thin neural tissue, a few hundreds micrometers thick. It is composed of five cell types precisely arranged in three nuclear layers separated by two plexiform layers (figure 1.2). Relative to the center of the eye, the outermost nuclear layer contains the photoreceptor cells which absorb light and convert it into neural signals through phototransduction. The inner nuclear layer contains the bipolar cells which receive and transform the signals coming from the photoreceptors. At last, the innermost nuclear layer contains the ganglion cells which in turn receive and transform the signals from the bipolar cells. The final signals are sent through the axons of the ganglion cells which form the optic nerve. In addition to the bipolar cells, two other types of interneurons are located in the inner nuclear layer: horizontal cells and amacrine cells which provide lateral connections in the outer and the inner plexiform layer respectively. Note that displaced amacrine cells can also be found in the ganglion cell layer.

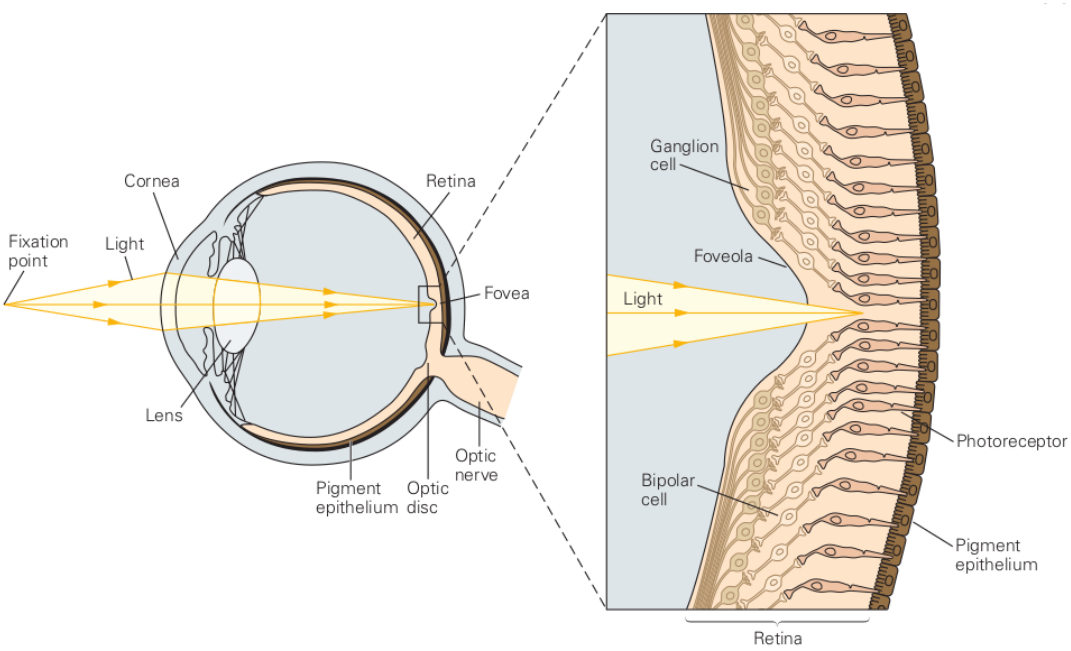

Figure 1.1: Anatomy of the eye (reprinted from (Kandel et al., 2000)). Light from a fixation point is refracted by the cornea and lens and focused onto the retina. Retina's output is then sent to the brain through the optic nerve. On a finer scale (such as the fovea, the very center of the gaze), light hits photoreceptors which convey visual information to ganglion cells trough bipolar cells. The axons of the retinal ganglion cells leave the eye cup through the optic disk to form the optic nerve.

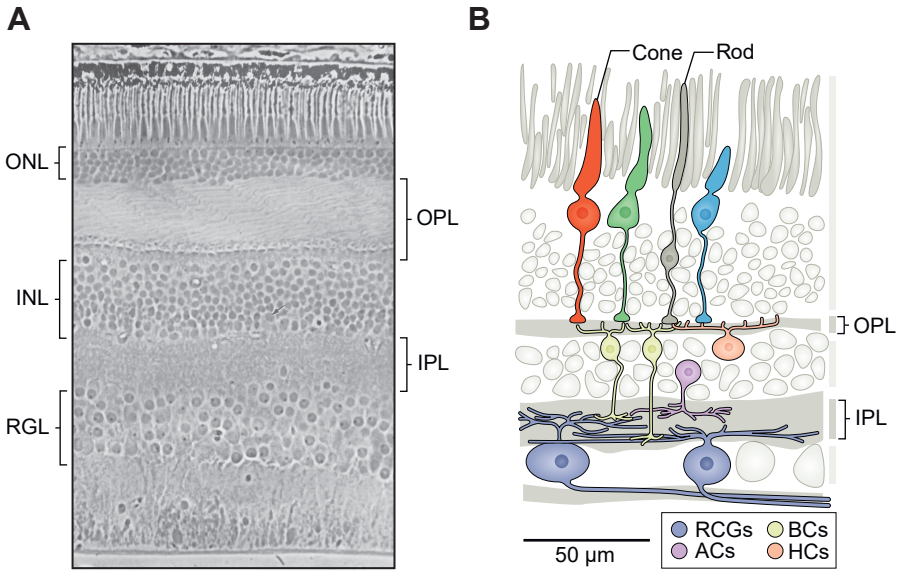

Figure 1.2: Vertical organization of the retina. **A.** Phase-contrasted image of a vertical section in the center of the fovea of a human retina. ONL: outer nuclear layer. INL: Inner nuclear layer. RGL: Retinal ganglion layer. (reprinted from (Boycott and Dowling, 1969)) **B.** Schematic of a vertical section in the periphery of a human retina. RGCs: Retinal ganglion cells. BCs: Bipolar cells. ACs: Amacrine cells. HCs: Horizontal cells. (reprinted from (Baden, Euler, et al., 2019)) **A-B.** OPL: Outer plexiform layer. IPL: Inner plexiform layer.

1.1.2 Photoreceptors sample the visual image

The photoreceptor cells have a specialized region, called the outer segment, which contains a light-transducing apparatus. There, absorption of photons sets in motion the transduction cascade which can ultimately hyperpolarize the cell. This decrease in membrane conductance slows the release of neurotransmitter (i.e. Glutamate) at the synaptic terminal, which is the neural signal transmitted to bipolar cells and horizontal cells.

There are two types of photoreceptors, rods and cones, distinguished by their morphology and their function. Rods have a long and cylindrical outer segment. They are associated with scotopic vision (e.g. dim light) because they are highly sensitive to light such that they are even able to detect single photon (Foster Rieke and Baylor, 1998). Cones have a shorter and conical outer segment, and are only involved in photopic vision (e.g. day light). Vertebrates present up to five types of cones distinguished by their sensitivity spectra (Baden, Euler, et al., 2019). For example, primates have L, M and S cones which respond to long-wave, medium-wave and short-wave respectively, and form the basis of color vision.

There are approximately 100 millions rods and 6 millions cones in the human retina, i.e. 17 rods for each cone. However, their spatial distributions are different with the highest density of cones found in the fovea and decreasing towards the periphery, whereas the highest density of rods is found around the fovea and decreases both towards the periphery or the center of the fovea.

The photoreceptor spacing limits the sampling resolution of the visual image. Interestingly, studies reported that the optical quality of the eye along its optical axis, in particular the diffraction due to the pupil, imposes a slightly stronger limit (Curcio and Hendrickson, 1991). This suggest that photoreceptors sample light patterns in an optimal way.

1.1.3 Ganglion cells transmit the neural image

The ganglion cells are located in the innermost nuclear layer of the retina. All their axons gather together at the optic disc to get out of the eye and form the optic nerve. Typically, brief electrical impulses known as action potentials or spikes are conducted by these nerve fibers towards the axon terminals. Together, these signals form the neural image which is sent to the brain.

To produce this neural representation of the visual image, ganglion cells selectively take their input neural signals from bipolar cells and amacrine cells to produce diverse response types. The very first functional distinction

between ganglion cells has been proposed by H. K. Hartline (1938) based on their responses to changes in light intensity. Some cells fire more spikes in response to light increments while others to light decrements. They are referred as ON cells and OFF cells respectively. Note that there also exist ON-OFF cells which respond to both increments and decrements of light.

The use of a localized stimulation such as a small light spot revealed an additional spatial organization (figure 1.3). Typically a ganglion cell is sensitive to light projected on a limited region of the retina, called the receptive field of the cell. This region present two areas where the effect on the responses are reversed: the center and the surround. For a typical ON cell, the response increases when a small light spot hits its center and decreases when it hits the surround. The highest increase in the response is seen when the spot entirely shines the center. When the spot covers both the center and surround, a weaker response is observed. It is possible to reach an even higher response by increasing the light in the center of the receptive field while decreasing the light in its surround. Note that the opposite is also true for OFF cells. This antagonistic center-surround spatial organization of the receptive field is particularly well suited to signal spatial contrast.

Functional classification of ganglion cells is not limited to this ON/OFF distinction. For example, the temporal dynamic of the responses can be either transient or sustained, meaning that the steady firing rate reached after a prolonged light stimulation is equal to the resting firing rate or to another firing rate respectively (Cleland et al., 1971). The full characterization of functional types is still an active debate. For example, more than 30 different types have been reported in the mouse retina (Masland, 2012; Baden, Berens, et al., 2016) which have been linked to morphologically and genetically defined populations of cells such as alpha cells (Van Wyk et al., 2009), ON direction-selective cells (W. Sun et al., 2006), JAM-B cells (I.-J. Kim et al., 2008), W3 cells (Y. Zhang et al., 2012) and OFF suppressed-by-contrast cell (Tien et al., 2015). Their spatial distribution is not uniform, W3 cells are one of the most numerous and have therefore a higher resolution. Similar observations have been made for other mammals (Dacey, 1993; Saito, 1983; Enroth-Cugell and Robson, 1966). More details about the complex functions implemented by ganglion cells will be given later in this chapter.

Around 1 million ganglion cells are present in the human retina (Watson, 2014). They are 100 times less numerous than the photoreceptors, which means that there is a strong compression of the neural image between these two populations. Since there is no feedback from the brain, the retina autonomously processes the visual information and all the computations are carried by the retinal network.

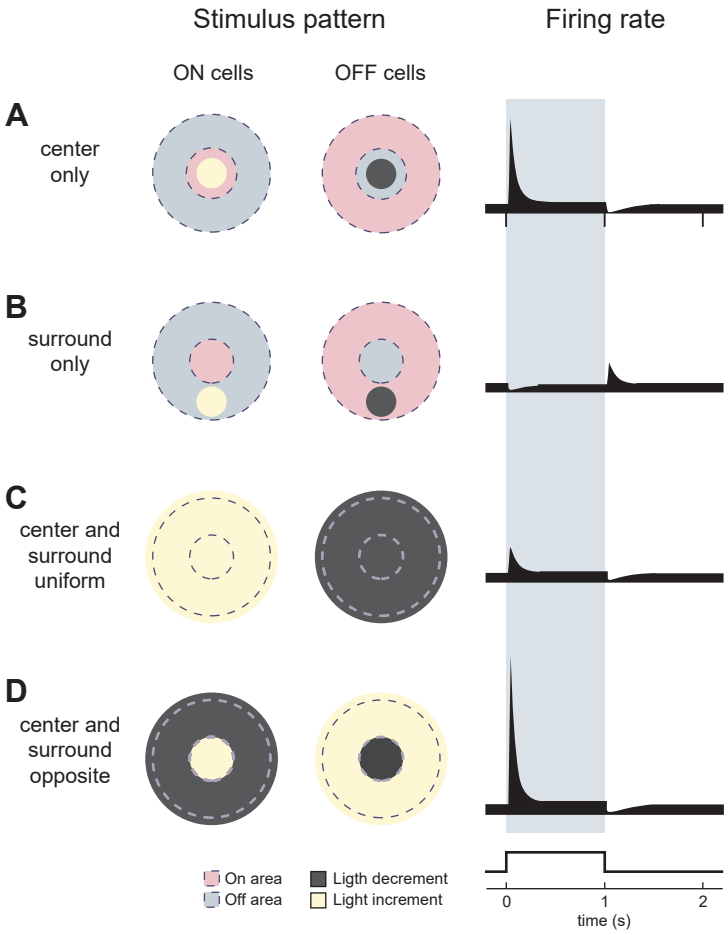

Figure 1.3: Antagonistic receptive field structures of ON and OFF cells (reprinted from (Kandel et al., 2000)). Idealized experiments where the stimulus are spots of either light increments (yellow) or light decrements (black), and receptive fields are decomposed in a ON area (red) and OFF area (blue). **A.** ON cells are excited by a light increment in their center, OFF cells by a decrement. Their firing rate increases at the onset of the stimulus, and decreases at its offset. **B.** They are suppressed if stimulated in their surround. **C.** Simultaneous stimulation of their center and surround elicits a similar but lower response compared to that of the center only. **D.** Stimulation of their center combined with the opposite stimulation of the surround elicits a similar and higher response.

1.1.4 Interneurons shape the neural image

Horizontal cells are the outermost interneurons, have wide dendrites, and form synapses with rods or cones. They receive the signals coming from neighboring photoreceptors, and are electrically coupled with other horizontal cells through gap junctions. Thus, they integrate the visual image over a receptive field which is larger than the photoreceptors, and provide them an inhibitory feedback proportional to the local mean luminance. In addition to this regulatory role, lateral inhibition also contributes to the surround suppression of ganglion cells. In fact, bipolar cells, which took their input from both photoreceptors and horizontal cells, already present receptive fields with an antagonistic structure.

They are 11 types of bipolar cells in the human retina, 10 cone bipolar cells and only 1 rod bipolar cell, due to the later apparition of rods during evolution (Baden, Euler, et al., 2019). Bipolar cells can also be distinguished from their response to glutamate releases from photoreceptors: OFF bipolar cells depolarize and ON bipolar cells hyperpolarize. These two classes correspond to a complex stratification in the inner plexiform layer such that ON bipolar cells connect to ON ganglion cells and OFF bipolar cells to OFF ganglion cells. In fact, synaptic connections are distributed in at least 10 specific sublayers (S. M. Wu et al., 2000; Roska and Werblin, 2001). In other words, ON and OFF pathways are already clearly established in the bipolar cell layer.

Amacrine cells are the most diverse with up to 50 morphological types (MacNeil and Masland, 1998). They are inhibitory neurons, usually axonless, which interact with ganglion and bipolar cells. They receive inputs from bipolar and other amacrine cells. Like horizontal cells, amacrine cells act laterally but are more specialized. For examples, starburst amacrine cells are important in the computation of direction-selectivity (Euler et al., 2002), and AII amacrine cells are essential for night-vision but is also recruited in day-vision (Mills and Massey, 1995). More generally, the electrical coupling observed for some types allow them to receive inhibition driven by distant bipolar cells, similar to the lateral inhibition of photoreceptors. This inhibitory signals contribute substantially to the center-surround organization of the receptive field of retinal ganglion cells. Thus, amacrine cells play a crucial role in the retina, even if they are hard to characterise due to their diversity and complexity in morphology, connections, functions.

1.2 Computational models of retinal functions

Since retinal ganglion cells were first recorded by Kuffler (1953) and Barlow (1953), a fundamental goal of sensory neuroscience has been to build accurate neural encoding models of ganglion cells. These computational models aim to provide quantitative predictions of the responses to various visual stimuli and most importantly to synthesize current understanding of retinal functions. Progressive discovery of the diversity of ganglion cell responses and functions yield many computational models of increasing complexity. Ultimately those models would give a detailed description of ganglion cell processing. This section starts with the description of idealized and simple computational models before describing specific ganglion cell types which need more complex models. The last part focus on the challenges posed to these models when used with more naturalistic models.

1.2.1 Spatiotemporal receptive field and linear models

The simplest computational description of ganglion cells relies on experimental mappings of their receptive fields. Early mapping techniques used a spot smaller than the whole receptive field to probe the different regions (H. K. Hartline, 1940; Kuffler, 1953; Rodieck, 1965; Daw, 1968) (now known as the "sparse noise" method), or spots of increasing size centered on the receptive field to reach the saturation of the response (Wiesel, 1960; Enroth-Cugell and Lennie, 1975; Peichl and Wässle, 1979). An alternative technique based on drifting gratings have been developed for rapid measurement of spatial receptive fields of multiple cells simultaneously (Enroth-Cugell and Robson, 1966). Nowadays, the most common technique relies on a spatiotemporal white noise stimulus (Sakai, Naka, and Korenberg, 1988; Korenberg et al., 1989; Sakai and Naka, 1992; Fred Rieke, Warland, et al., 1999; S. P. Brown et al., 2000; Chichilnisky, 2001) which has the advantage to reveal fine spatial and temporal structures. Recently, an additional method based on randomly flashing bars have been proposed, the filter back-projection technique (Johnston et al., 2014), which is faster than spatiotemporal white noise at estimating the basic shape of the receptive fields.

All these methods provide a spatiotemporal description of receptive fields. If linearity is assumed between the stimuli and the responses, computations of ganglion cells can be modeled as spatiotemporal integrations of the stimulus. Rodieck and Stone (1965a) was the first to devise such a model. Given a full spatiotemporal description of the visual stimulation of the retina $s(x, y, t)$,

the firing rate of a ganglion cell $\lambda(t)$ at time t is modeled as:

$$\lambda(t) = \iiint f(x, y, \tau) s(x, y, t - \tau)\, dx\, dy\, d\tau \tag{1.1}$$

where $f(x, y, \tau)$ represents the spatiotemporal receptive field of the cell (i.e. a linear filter). For simplification, a common assumption is space-time separability such that this filter can be decomposed in two terms: $f(x, y, \tau) = f(x, y)h(\tau)$ where f correspond to the spatial profile and f to the temporal profile. Rodieck and Stone (1965a) provided mathematical expressions to describe idealized profiles. The idealized temporal kernel is characterized by a transient change with exponential decay:

$$h(\tau) = a\, \mathrm{e}^{-\tau/\tau_d} \quad \text{if } \tau \geq 0 \text{ else } 0 \tag{1.2}$$

where a represent the magnitude of the change and τ_d the time constant of the decay. And the idealized spatial kernel is assumed to have perfect radial symmetry such that the spatial filter can be expressed as $f(x, y) = f(r)$ where $r^2 = x^2 + y^2$, and thus can be simply characterized as a difference of Gaussians:

$$f(r) = a_c\, \mathrm{e}^{-r^2/r_c^2} - a_s\, \mathrm{e}^{-r^2/r_s^2} \tag{1.3}$$

where a_c and a_s corresponds to the amplitudes of the center and surround, and where r_c and r_s represents their radii. Typically, these values are chosen such that $a_c > a_s$ and $r_c < r_s$ to specify a small and strong center and a large and dim surround. Slight modifications and improvements are often necessary to match actual responses such as the addition of spontaneous firing rate, the addition of a sustained component in the temporal profile, or the rectification of negative predictions of the firing rate.

This model was found useful to predict quantitatively the firing rate of cat retinal ganglion cells in response to moving shapes of different sizes such as moving bars (Rodieck and Stone, 1965a; Rodieck and Stone, 1965b). However some assumption doesn't hold when confronted with experimental data. For example, the space-time separability is often unsatisfied with response to light falling in the surround delayed in comparison with the response to light falling in the center. The main reason is that it recruits two different pathways. In case of a surround stimulation, the pathway is longer and the neural signal flows through horizontal or amacrine cells and passes across additional synapses (Benardete and Kaplan, 1997; Enroth-Cugell and A. W. Freeman, 1987; Sakai, J. Wang, et al., 1995). Another issue is the linearity of the responses predicted by this model. If the magnitude of the stimulus is multiplied by a given factor then the predicted firing rate is also multiplied by this factor. This is usually not the case since neurons can't fire an

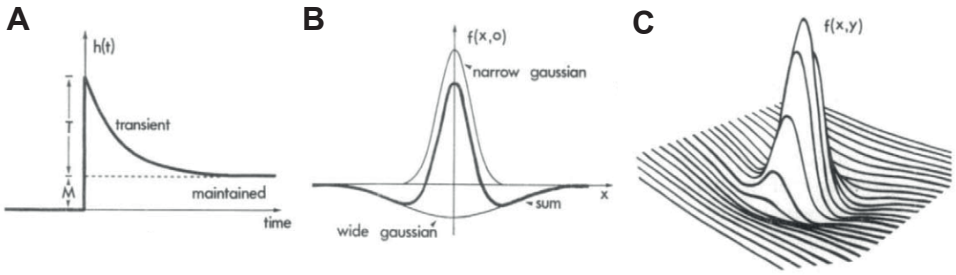

Figure 1.4: Idealized receptive field as a difference of Gaussians (reprinted from (Rodieck and Stone, 1965a)). **A.** Idealized temporal kernel with an exponential decay. **B.** Idealized spatial profile as a difference of Gaussians. **C.** Idealized 2D spatial kernel.

unlimited number of spikes in a limited time period. Nevertheless, such a linear relation have been observed when the light modulations remain small compared to the mean luminance (Benardete and Kaplan, 1997; Sakai, J. Wang, et al., 1995). The assumption of a Gaussian shape for the spatial receptive field is also problematic. Fine-scale structure have been measured for ganglion cells in primate (Passaglia et al., 2002), mouse (G. W. Schwartz et al., 2012) or salamander (Soo et al., 2011). In fact, G. W. Schwartz et al. (2012) found that the fine-scale organization of the dendritic arbor can be mapped to these fine-scale irregularities.

1.2.2 Pseudo-linear models

Pseudo-linear models are the prevalent models of light responses and solve the main issues of the model presented in the previous section. For example, the nonlinear (LN) model consists of a spatiotemporal filter which describes the integration of the stimulus over space and time, and a nonlinear activation function which maps the scalar output of the filter to the predicted instantaneous firing rate. Two techniques are used to fit the parameter of the LN model: estimation of the spike-triggered average with spatiotemporal white noise stimulation (Brenner et al., 2000; Chichilnisky, 2001; O. Schwartz et al., 2006), or information-maximization or maximization of likelihood (Paninski, 2004; Sharpee et al., 2004).

The LN model can predict the responses to white noise stimulation of some ganglion cells in the salamander and macaque retina (Chichilnisky, 2001). Surprisingly, Bomash et al. (2013) showed that responses to natural

scenes can also be predicted in the mouse retina. However, this is rather the exception than the rule and this pseudo-linear model is in practice unable to predict responses for many combinations of stimuli and cell types.

Another example of pseudo-linear model is the generalized linear model (GLM) (Pillow, Shlens, Paninski, et al., 2008) which has several conceptual and technical advantages over the LN model and other pseudo-linear models, such as a noise model which accounts for bursting and refractoriness (Berry and Meister, 1998), cross-coupling between cells to capture correlations present in populations (Pillow, Shlens, Paninski, et al., 2008), and robust estimation of the parameters through log-likelihood maximization (Paninski, 2004).

GLM are also efficient to describe responses to artificial and stochastic stimuli from macaque ganglion cells (Pillow, Shlens, Paninski, et al., 2008). Nevertheless, this success may be partly due to the low spatial resolution (pixel size of 120 µm) of the white noise stimulation. This imposes an implicit assumption that signals from photoreceptors are linearly integrated over space. J. Freeman et al. (2015) showed that this assumption does not hold with stimuli of higher spatial resolution (pixel size of 3.4 µm). Heitman et al. (2016) confirmed that the GLM can predict the responses to white noise stimulation with medium resolution. However, they also showed that the GLM was not able to generalize to predict responses to naturalistic stimuli. This was also the case when the model was trained and tested with the naturalistic stimuli. This finding suggests that additional computational mechanisms are necessary to predict responses to natural scenes.

1.2.3 More complex computations and models

Pseudo-linear models carry a simplified view of the retina's function. They characterize each ganglion cell as a simple filter which enhance local contrast in space and time, combined with temporal mechanisms which induce refractoriness, burstiness and adaptation. Nevertheless, this view fails to provide any explanation to the diversity of cell types or the complexity of the connection patterns encountered in the retina. The idea that each ganglion cell type computes something rather specific to the visual scene is now well established (Meister and Berry, 1999; Roska and Werblin, 2001; Wässle, 2004; Gollisch and Meister, 2010; Silveira and Roska, 2011). This is supported by detailed characterization of specific ganglion cell types which are described in this section. A large amount of work have been dedicated to nonlinear modelling of these specific cells. An extensive review of the corresponding studies is out of scope for this section. Instead, references to relevant nonlinear models will be introduced when necessary.

Direction-selective ganglion cells

Direction-selective ganglion cells (DSGCs) respond strongly to moving stimuli, such as moving bars, but only for a specific direction of motion (Barlow et al., 1964; Taylor and Vaney, 2003; Demb, 2007). There are different types of DSGCs distinguished by the polarity of their responses to light steps: ON, OFF or ON-OFF. ON-OFF DSGCs detect motion in one of the four cardinal directions while ON DSGCs detect movement in the dorsal, ventral and nasal directions (Oyster and Barlow, 1967). OFF DSGCs have been discovered later and detect upward motion in the visual scene (I.-J. Kim et al., 2008). Note that a small movement of one tenth of the receptive field is sufficient to elicit a response from a DSGC (Vaney et al., 2012). ON-OFF DSGCs respond to a wide range of velocities whereas ON DSGCs respond only to slow motions (Vaney et al., 2012). Overall direction-selective cells are one of the typical example of highly nonlinear computations which occur in the retina and need dedicated modeling effort (e.g. Reichardt-Hassenstein model (Hassenstein and Reichardt, 1956; Borst and Euler, 2011)).

Orientation-selective ganglion cells

Orientation-selective ganglion cells are selective for either horizontal or vertical bars (stationary or moving). They are characterized by a total absence of response for bars presented orthogonally to their preferred direction. First observed in the rabbit retina (Levick, 1967), they have also been found in mouse retina (Nath and G. W. Schwartz, 2016; Baden, Berens, et al., 2016). A model in which a circular excitatory center is combined with a larger but oval inhibitory surround is able to capture responses (Caldwell et al., 1978).

Suppressed-by-contrast ganglion cells

Suppressed-by-contrast (SbC) ganglion cells present a spontaneous firing rate and reduce their activity in response to light increments (ON) and decrements (OFF). They have been observed in the rabbit (Levick, 1967), cat (Rodieck, 1967), macaque (De Monasterio, 1978) and mouse (Tien et al., 2015) retinas. The suppressive spike responses of SbC RGCs is linked to saccade-like eye movements and blinks (Tien et al., 2015). Global image changes such as shifts of random textures or blackouts elicit a strong inhibition of these cells. This suggests that they indicate self-generated stimuli by eye movements and blinks and may signal that other ganglion cells switch circuits to process the refreshed visual scene.

X and Y cells

Using fine gratings, Enroth-Cugell and Robson (1966) found that some ganglion cells, called the X cells, don't elicit any response when there is a simultaneous increase of the luminance in half of the receptive field and a decrease in the other half (figure 1.5A). This was the case both for the center and the surround, and suggested linear integration of the spatial components of the stimulus in these areas. However, responses were elicited for other cells, called the Y cells (figure 1.5B), which suggested nonlinear spatial integration in this case. The spatial frequency of the grating needed to distinguished X and Y cells has to be smaller then the center of the receptive field (Demb et al., 1999). Y cells are also selective to moving textured pattern, whatever the texture and direction of motion are (Demb et al., 1999; Petrusca et al., 2007). Overall X and Y cells are the examples that the assumption of linear spatial integration doesn't hold for all ganglion cells. Thus classical linear and pseudo-linear models are not able to capture responses of Y cells. Instead a specific model, called the subunit model, is needed (Hochstein and Shapley, 1976; Victor and Shapley, 1979; Victor, 1987). It gives quantitative predictions of the responses to reversing gratings (Enroth-Cugell and A. W. Freeman, 1987; Ölveczky et al., 2003). The subunit model can also accurately predict responses to white noise stimuli and its inferred components reflect the anatomy of the underlying neural circuitry (J. Freeman et al., 2015). Note that different types of spatial integration can also be found within the same cell with, for example, a linear integration in the center and a nonlinear one in the surround (Deny et al., 2017).

Other ganglion cell types

Many other ganglion cell types which correspond to complex and nonlinear computations have been observed. Object motion-sensitive (OMS) cells are able to detect moving objects in the visual scene (Lettvin et al., 1959; Ölveczky et al., 2003). They detect differential motions between small parts of the visual image and the background. These ganglion cells elicit a strong response when the apparent motion in the center of the receptive field is different from the motion in the surround. Note that they are really different from direction-selective cells which rather detect absolute motion than differential motion. The underlying circuits have been identified (Ölveczky et al., 2003; Ölveczky et al., 2007; Baccus, Ölveczky, et al., 2008) and synthesized in a subunit model which combined an excitatory signal from the center of the receptive field and inhibitory signals coming from other motion-sensitive cells in the neighborhood.

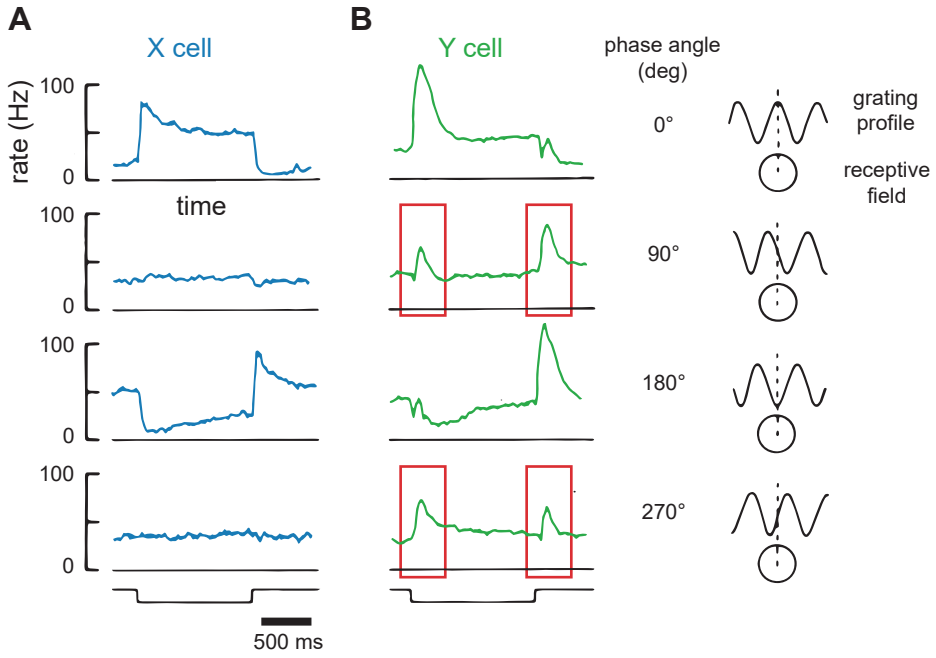

Figure 1.5: Responses of X and Y cells to grating patterns with different spatial phases (reprinted from (Enroth-Cugell and Robson, 1966)). A sinusoidal grating pattern was introduced and withdrawn as indicated by the lowest traces. The different lines of the plot correspond to different spatial shifts of the grating relative to the mid point of the receptive field center as depicted on the right of the figure. **A.** Responses of an OFF-center X cell to spatial gratings (spatial frequency: 0.13 cycle/deg). **B.** Responses of an OFF-center Y cell to spatial gratings (spatial frequency: 0.16 cycle/deg). This cell responds to spatial shifts for which the grating patterns lie with odd symmetry about a diameter of the receptive field, and for which the changes in luminance over one half of the receptive field are the exact inverse of the changes over the other half (framed in red). This indicate nonlinear spatial integration over the receptive field.

Similarly, approaching motion-sensitive ganglion cells have responses evoked preferentially by expanding bars which mimic approaching motion such as looming predators (Münch et al., 2009). In addition to vertical and horizontal motions, they provide motion detection along the third dimension. The circuits that implement this function use the same strategy as the OMS cells with inhibitory signals which suppress responses to the non-preferred motion direction (Münch et al., 2009).

As a last example, there are also intrinsically photosensitive ganglion cells in the retina (Provencio et al., 2000; Hattar et al., 2002). These ganglion cells can transduce light into electrical signals like photoreceptors. They regulate non-image forming functions such as the photoentrainment of the circadian rhythms, the control of pupillary light reflex or sleep. Multiple subtypes have been identified (Tu et al., 2005) which suggest several specific computations.

For these complex cell types, nonlinear modeling is necessary to provide quantitative predictions of their responses and to synthesize current understanding of their functions. Nonlinear models have also been used to study ganglion cells processing with less importance given to their functional types but more to the recruited retinal pathways. For example, the first spike latency after the onset of a flashed stimulus has been modeled with a subunit model which recapitulate the convergence of the On and Off pathways on ganglion cells (Gollisch and Meister, 2008b; Gollisch and Meister, 2008a).

1.2.4 Computations in naturalistic contexts

The computation of the different ganglion cell types mentioned previously have been heavily studied with engineering tools (Rowe and Stone, 1980). Artificial stimuli have been specifically designed to probe specific function of the retina. In many cases, an antagonistic structure of the estimated receptive fields was sufficient to explain the corresponding function. However the choice of the stimulus can heavily conditioned the functions which are recruited and active. For example, OMS ganglion cells well are characterized by receptive fields with a center-surround organization when probed with a spatiotemporal white noise. The detection of moving object, the specific function of these cells, is unveiled when the appropriate stimulus is used such as an image patch moving in the center of the receptive field while the background remains static. When chosen carefully, artificial stimuli have been particularly useful to investigate and quantify isolated function of some ganglion cell types. Nevertheless, these artificial stimuli are quite different from the stimuli which are actually encountered in a natural environment. In this context, multiple functions may be active at the same time and compete to relay the most relevant visual information to the brain.

Adaptations to low order statistics

The retina operates under many different lightning conditions. Light intensities vary over 10 orders of magnitude in a day (Sakmann and Creutzfeldt, 1969). In comparison, the dynamic range of the responses of ganglion cells spans only 2 orders of magnitude. Thus, ganglion cells can not transmit all the information but need to modulate their inputs and adapt their functions depending on the mean luminance of the visual scene (figure 1.6). This modulation is referred to as light adaptation (Fred Rieke and Rudd, 2009) and affects the receptive field properties with. For example, the suppressive surround becoming weaker in dim light (Farrow et al., 2013).

Another form of adaptation concerns the contrast of the visual scene, the standard deviation of the luminance distribution. This modulation have been observed by Smirnakis et al. (1997) in salamander and rabbit retinas. When an increase of contrast is encountered, retinal ganglion cells change their mean levels of activity. By doing so, they become less sensitive to the light variations and respond more rapidly over time. In fact two time scales have been distinguished (Baccus and Meister, 2002), with a fast change during the first 100 ms where the sensitivity decreases and the kinetics increases, and a slow change during multiple seconds where the sensitivity continues to decrease.

A more complex adaptation to pattern have also been observed (Smirnakis et al., 1997). When a new spatiotemporal pattern with same mean luminance and contrast as the previous one is encountered, ganglion cells change their mean levels of activity. By doing so, they become less sensitive to the pattern over time. For example, Hosoya et al. (2005) used horizontal and vertical bars to trigger pattern adaptation in salamander and rabbit retinas.

Response model for high order statistics

Natural scenes are not only characterised by their mean luminance and contrast but have considerable structure which correspond to many high order statistics (Simoncelli and Olshausen, 2001). The statistical redundancy of natural images have been pointed out by multiple authors (Attneave, 1954; D. J. Field, 1987; Ruderman and Bialek, 1994).

Does the retina take advantage of the correlations in natural scenes? Srinivasan et al. (1982) proposed that lateral inhibition found in the retina might be a way to reduce spatial redundancies (Barlow, 1961), and provided the first quantitative evidence for decorrelation in the compound eye of the fly. This problem has also been investigated in the presence of noise (Atick

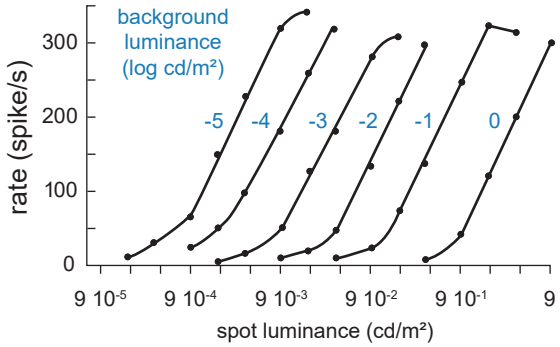

Figure 1.6: Light adaptation in a cat's retinal ganglion cell (reprinted from (Sakmann and Creutzfeldt, 1969)). Responses of an ON-center ganglion cell to uniform illuminations of its receptive field for different steady background luminances. The peak firing rate is plotted against the logarithm of the spot luminance. Each curve corresponds to a different background luminance indicated in log cd/m².

and Redlich, 1992). More recently, the contribution of receptive field structure to the efficiency of the retinal code has been revised. In fact, most of the decorrelation was found to be accomplished by nonlinear processing (Pitkow and Meister, 2012) since linear receptive fields failed to decorrelate much. Note that this theoretical framework based on information theory can also describe the emergence of different types of ganglion cells (Kastner et al., 2015) which is controlled by the amount of noise affecting neural responses.

To reveal the feature selectivity of different ganglion cell types, complex naturalistic stimuli have been used such as natural movie scenes (Nirenberg et al., 2001), even with eye movements (Turner and Fred Rieke, 2016). However, the ganglion cell filters can't be estimated with classical techniques such as the spike-triggered analysis because of non-zero stimulus correlations. Some adaptations allowed to use generalized linear models to capture responses in the macaque retina with limited success (Heitman et al., 2016). Another tool, called maximally informative dimensions (MID), has been developed to search iteratively in the stimulus space the dimensions that maximize the mutual information between the spike train and the stimulus (Sharpee et al., 2004). However this method requires a lot of data and there is no guarantee that the search procedure will find the optimal solution because of local minimums. More recently, deep learning has been employed to build models of retinal ganglion cell responses to natural scenes (McIntosh et al.,

2016; Maheswaranathan, McIntosh, Kastner, et al., 2018). This approach seems a promising solution to build a quantitative and interpretable model of the retina which generalize to naturalistic stimuli.

Chapter 2

Spike sorting review

We have seen that the retina is a complex sensory system where ganglion cells encode visual information. The flat structure of the retina makes these cells particularly accessible for recordings with microelectrode arrays to access their spiking activity. Over the last decades, technological advances improved the sampling resolution of these devices. The spacing between electrodes decreased from hundreds of microns down to tens of microns.

This improvement makes the voltage signals recorded from nearby electrodes less independent. A spike from one cell is detected on several electrodes and conversely one electrode detect spikes from multiple cells. This cross-talk between neurons and electrodes challenges the traditional spike sorting methods used to reconstruct the spiking activity of each neuron. These methods generally fail to identify synchronous or near-synchronous spikes whose voltage waveforms overlap (Pillow, Shlens, Chichilnisky, et al., 2013).

Here, I review new algorithms used to recover the precise activity of individual neurons from extracellular voltage traces recorded with dense and large-scale microelectrode arrays. I show that they follow the same strategy even if they have been developed independently, I present their main differences, and I discuss the issues that need to be resolved by future spike sorting methods.

This chapter corresponds to a review which was previously published as:

Baptiste Lefebvre, Pierre Yger, and Olivier Marre (2016).
Recent progress in multi-electrode spike sorting methods.
Journal of Physiology-Paris, 110(4), 327-335.

Abstract

In recent years, arrays of extracellular electrodes have been developed and manufactured to record simultaneously from hundreds of electrodes packed with a high density. These recordings should allow neuroscientists to reconstruct the individual activity of the neurons spiking in the vicinity of these electrodes, with the help of signal processing algorithms. Algorithms need to solve a source separation problem, also known as spike sorting. However, these new devices challenge the classical way to do spike sorting. Here we review different methods that have been developed to sort spikes from these large-scale recordings. We describe the common properties of these algorithms, as well as their main differences. Finally, we outline the issues that remain to be solved by future spike sorting algorithms.

2.1 Introduction

Progress in neuroscience relies to a large extent on the ability to record simultaneously from large populations of cells, in order to understand how information is represented among neurons. One of the most popular techniques to measure such an activity is the use of arrays of extracellular electrodes. With these devices, each electrode records the extracellular field in its vicinity and can detect the action potentials emitted by the neighboring neurons. In contrast to intracellular recording, those extracellular recordings do not give a direct access to the neuronal activity: one needs to process the recorded signals to extract the spikes emitted by the different cells around the electrode. This process is termed *spike sorting*, and many algorithms have been suggested to do it efficiently (see Lewicki, 1998 or Rey et al., 2015 for a review).

The first extracellular recordings were performed with a single electrode, and could only give access to 3-5 neurons (Gerstein and Clark, 1964). A recent study (Pedreira et al., 2012) highlighted that the maximal number of accessible neurons should lie between 8 and 10 in that case. Over the last decades, there has been a strong effort to increase the number of electrodes, and therefore the number of recorded neurons. Spike sorting algorithms had to be adapted to process this increasingly large amount of data. At first, electrodes were spaced by hundreds of microns such that the spike of one cell could only be detected on a single electrode (Jones et al., 1992; Shoham et al., 2003). In that case, spike sorting on a large amount of electrodes could simply be done by processing each electrode independently. The parallelization of the problem for large amount of independent electrodes was relatively easy

to address.

However, devices where electrodes are packed with a high density have also been developed. The spacing between electrodes is much smaller (tens of microns). As a consequence, a spike from a single cell can be detected on several electrodes. Conversely, each electrode will detect the activity of many cells, a property already encountered in the case of single electrode. This increased density helps a lot to resolve single cells (Gray et al., 1995; Franke, Pröpper, et al., 2015), but electrode signals could not be processed independently. Spike sorting algorithms had to be adapted to this new type of data. While for small numbers of electrodes (e.g. tetrodes), methods that could be seen as adaptations of single electrode sorting worked very well (McNaughton et al., 1983; Harris et al., 2000; Gao et al., 2012), this is not the case with new devices designed with hundreds of electrodes all densely packed. CMOS-based devices with thousands of electrodes have been tested and are now frequently used (Berdondini et al., 2005; Fiscella et al., 2012; Müller et al., 2015; Hilgen et al., 2016), calling for new algorithmic methods, largely different from the usual sorting methods.

Here we review the different spike sorting algorithms that have been proposed to process recordings from these novel high-density devices. We will first explain the limitations of classical spike sorting approaches to process these large-scale, dense recordings. Then, we will outline the main changes introduced by these new algorithms compared to classical spike sorting approaches. We will emphasize that most of these new methods follow the same global strategy, although they have been developed independently by different groups. Therefore, we will outline the common properties shared by these algorithms, before explaining and discussing their main differences. Finally, we will discuss the issues that still need to be resolved by future spike sorting algorithms.

2.2 The challenge posed by large-scale multi-electrode recordings to classical approaches

Most of the classical approaches to spike sorting can be decomposed in two main steps. First, some specific features of the spike waveforms are extracted from the raw data. This allows each spike to be characterized by a small set of features. Using these features, each spike can now be seen as a point in a low dimension space, and the second step consists in clustering all the points in this reduced space.

For the first step, earliest methods only extracted the spike amplitude (Hubel, 1957), and width (Meister, Pine, et al., 1994) of each spike. More recently, some methods use the full waveform directly when the number of electrodes remains small (Pouzat et al., 2002). Another standard technique is to project each waveform on a set of basis functions (Litke et al., 2004; Quiroga, Nadasdy, et al., 2004), that are either found by performing a principal component analysis (PCA) on the entire set of waveforms (Egert et al., 2002; Pouzat et al., 2002; Einevoll et al., 2012; Swindale and Spacek, 2015), or by choosing a wavelet basis (Letelier and Weber, 2000; Hulata et al., 2002; Quiroga, Nadasdy, et al., 2004). For a comparison between PCA and wavelet based analysis, see (Pavlov et al., 2007). Note that the two can be combined (Bestel et al., 2012).

Once the dimensionality has been reduced, to tackle the problem of the clustering step, several approaches have been used, but the most standard approach is to fit the clusters with a mixture of Gaussians (Wood et al., 2004; Rossant, Kadir, et al., 2016; Kadir et al., 2014). However, one could also find in the literature approaches such as paramagnetic clustering (Quiroga, Nadasdy, et al., 2004), mean-shift clustering (Swindale and Spacek, 2014) or even k-means clustering (Atiya, 1992; Chah et al., 2011). Another interesting approach is to consider the most consensual clustering across an ensemble of k-means solutions (Fournier et al., 2016).

Not all standard methods strictly follow this workflow. For example, linear filtering is an alternative approach which identifies the optimal linear filter to distinguish one signal, of unknown temporal position but of known waveform, from a finite number of other signals of known waveforms, observed on noisy electrodes. This approach was first proposed by Roberts and D. K. Hartline (1975), then by Gozani and Miller (1994) and more recently by Franke, Natora, et al. (2010). This method is similar to template matching approaches that we will describe later. An alternative approach is Independent Component Analysis (ICA) where the first step demixes blindly the data and extract the individual source signals from which spikes are identified (Takahashi et al., 2003; G. D. Brown et al., 2001; Jäckel et al., 2012). Note that variants, such as the convolutional Independent Component Analysis (cICA) of Leibig et al. (2016), has been developed. However, there is no guarantee that the independent components found by those algorithms are indeed isolated neurons.

While all of these methods can be successful when one electrode captures the signals from a only few cells, and when one cell is only recorded by one or a small number of electrodes, it is not trivial to scale them up to process a large number of densely packed electrodes. In recordings performed by large and dense multi-electrode arrays, the spike waveforms live in a high

dimensional space, and this makes the clustering challenging. We will review below some suggested improvements to enable clustering on a large number of electrodes.

Finally, a more fundamental problem with clustering-based approach is that the extraction of features from one spike can be distorted by the presence of other spikes nearby. As a consequence, most of the overlapping spikes are not captured by clustering approaches, because they correspond to points in the feature space that are far from the centers of the corresponding clusters. This is a major challenge for clustering techniques (Bar-Gad et al., 2001), that we will explain in more details below. In large scale and dense multi-electrode recordings, overlapping spikes become the rule rather than the exception. Solving this issue is one of the motivation behind new algorithms, based on template matching, that we will review and discuss in a second part.

2.3 Improvements of the clustering

In order to be able to scale up and perform spike sorting for large number of channels with the classical algorithms mentioned above, several refinements of the clustering have been proposed by various groups.

2.3.1 Improved spike detection

Rossant, Kadir, et al. (2016) have proposed a method that pre-processes the data to make clustering easier for multi-electrode sorting. As explained above, the spike of a single cell can be detected on multiple electrodes. Conversely, spikes from several cells can be seen on the same electrode. They designed a flood fill method to group together spikes detected on different electrodes that correspond to a single cell. For this they connect together spikes detected synchronously on adjacent electrodes. The exact algorithm to connect adjacent events bears some similarity with standard image processing algorithms, like the Canny contour detection. Spikes are therefore defined as spatio-temporal events, with a given spatial extent, called a mask, for each of them.

In a second step, for each of these events, they remove any voltage deflection outside of the mask, and replace it with noise. This masking removed part of the distortion induced by other spikes when estimating the features, and improved the performance of the clustering. While this improvement is of great help for overlapping spikes that are distant enough in space, it is less clear how it will help for spikes coming from two cells that are physically close. In that case, some electrodes will detect spikes from the two cells, and

their masks will strongly overlap. Therefore this masking process may only help avoiding temporally overlapping spikes from distant cells.

2.3.2 Pre-clustering

Marre et al. (2012) and Swindale and Spacek (2014) use a method to break down the clustering problem into multiple smaller parts. After detecting all the spikes in the recording, waveforms are grouped in different subsets according to the electrode where the highest voltage peak was found. Instead of performing a single clustering algorithm on all the waveforms, this grouping outputs N subsets, if N is the number of electrodes. Each subset contains all the spikes peaking on the same electrode. A clustering is then performed on each of these subsets independently.

Note that this pre-grouping does not assume that the spikes are only detected on a single electrode, which would amount to multiple single electrode sorting. Here, after this pre-grouping, a clustering is performed for each group, and this clustering used the information available on all the electrodes. This simplification allows reducing drastically the number of spikes that have to be processed together. It also allows a simple parallelization of the clustering, which is crucial for large-scale recordings with hundreds or thousands of electrodes.

The main issue with this method is that a cell that is located between two electrodes might emit spikes that peak alternatively on one or the other electrode. In that case, the cell will be split between two different groups, and subsequently in two different clusters. This strategy has therefore to be combined with a later step where all the clusters that correspond to the same cell are merged together. This method is therefore on the side of overclustering the spikes, and merging the different clusters later on. However, merging clusters is usually easier than splitting them since there is one possible result for the first operation whereas the second one presents many possible solutions.

2.3.3 Main issues associated with clustering

A complete review of all the clustering algorithms used for spike sorting is beyond the scope of this review. However, we would like to outline the main issues associated with the clustering step, that are common to almost every clustering algorithm.

Mathematical definition and non-linear optimization

Two of the main issues associated with any spike sorting solution relying on a clustering approach can be found in the roots of the clustering *per se*. Mathematically, the clustering suffers from a lack of problem statement and problem resolution. First, one need to agree on a mathematical definition of the notion of cluster to state the problem. Because there exists many different cluster models (e.g. centroid models, distribution models, density models), there are numerous notions of what a cluster is. It is not obvious if one of these notions would fit appropriately to the biological reality. Hence, the first problem is that the stated problem is an approximation of the true problem. Thus, the solution to this clustering problem is an approximated solution to the true problem. This is why it often requires the user to spend a rather large amount of time in manual curation, because the solution to the true problem is in the neighborhood of the approximated solution.

Second, solving a clustering problem brings additional issues. The different methods used to do clustering involve finding the minimum of an objective function, and the solution landscape almost surely presents local minima. As a consequence, running twice the same clustering algorithm with two different set of parameters (i.e. internal parameters such as initial centroids for the k-means algorithm) can lead to different results. The reason is that the two runs can be trapped in two different local minima. In many cases it takes several trials before converging to the global minimum, which increases the computational cost. In practice, the algorithm may stop before convergence because the more complex/challenging is the solution landscape, the less likely is the convergence in a reasonable time.

Overlapping spikes

More importantly, as mentioned above, a major issue with clustering is that it will miss many overlapping spikes. If two spikes are overlapping on the same electrode, there will be a distortion in the feature estimation, that will drive the spike beyond the limits of the cluster defined on isolated spikes. Note that the superposition problem has been known for a long time (Prochazka et al., 1972; Roberts and D. K. Hartline, 1975). The issue was apparent in Harris et al., 2000: they showed that the error rate of the spike sorting is strongly increased during spindle waves, which are epochs of synchronous firing. False positive errors could change from 5% to almost 80%, and false negative were also increased by at least 20%. The issue was more extensively studied by Pillow, Shlens, Chichilnisky, et al. (2013), where they show that synchronous spikes will be missed by a pure clustering approach. An addi-

tional study of Franke, Pröpper, et al. (2015) confirms that clustering-based methods perform poorly for overlapping spikes, as shown by Lewicki (1998) and Quiroga, Nadasdy, et al. (2004). Template matching approaches have been developed in order to deal with these overlapping spikes.

2.4 Template matching approaches

Several template matching approaches have been developed for spike sorting (Pillow, Shlens, Chichilnisky, et al., 2013; Pachitariu et al., 2016; Marre et al., 2012; Yger et al., 2016; Prentice et al., 2011). Note that historically the use of template matching (Gerstein and Clark, 1964) predates the use of clustering (Simon, 1965) and then experienced renewed interest. All these methods usually assume that the extracellular signal can be decomposed as a sum of so-called "templates" (one template is the average extracellular waveform triggered by one neuron) plus some noise:

$$\vec{s}(t) = \sum_{ij} a_{ij}\vec{w}_j(t - t_i) + \vec{e}(t) \tag{2.1}$$

where $\vec{s}(t)$ is the signal recorded over the electrodes of the multi electrode array and over multiple time points. $\vec{w}_j(t-t_i)$ is the spatiotemporal template associated with each cell, which represents the average waveform triggered on the electrodes by cell j (example in figure 2.1B). t_i are all the putative spike times over all the electrodes, a_{ij} is the amplitude factor for spike time t_i for cluster j, and $\vec{e}(t)$ is the background noise.

In this notation, the spike train associated with cell j is the set of times t_i where a_{ij} is different from zero. The template matching approach aims at finding the right values for $\vec{w}_j(t)$ and a_{ij}, i.e. to find where each cell spiked. Almost all the template-matching based methods try first to find the value of the templates, and then the values of a_{ij}. Depending on the algorithm, the amplitude values can only be 0 or 1, or can take any continuous value. We will review these methods in subsections 2.4.3 and 2.4.4.

2.4.1 Template extraction

To estimate the templates, most methods usually rely on clusters extracted from the recording using one of the methods described above. Each cluster corresponds to a set of snippets in the extracellular data. The snippets of a given cluster are supposed to be realizations of action potential of a single cell. We want to extract a canonical representative (i.e. a template). A naive method would be to consider the average waveform of these snippets.

However, because averaging is very sensitive to outliers, if some of the snippets also include overlapping spikes from other cells, they might distort the estimate of the template. Two solutions have been developed to circumvent this issue. The simplest (and fastest) one is to take the median at each time point instead of the mean (Marre et al., 2012; Yger et al., 2016). The median is way less sensitive to outliers than the mean. This method usually solves the issue of overlapping spikes.

Another solution is to model the extracellular signal from the clustering result:

$$\vec{s}(t) = \sum_{ij} b_{ij}\vec{w}_j(t - t_i) + \vec{e}(t) \tag{2.2}$$

Notations are similar to equation 2.1, except that b_{ij} are binary variables such that b_{ij} is set to 1 if t_i is associated to cluster j, and to 0 otherwise. Here the unknown variables are the templates $\vec{w}_j(\tau)$. Under these conditions, it is possible to find the templates that will fit the extracellular data best, by minimizing the following square difference (Pillow, Shlens, Chichilnisky, et al., 2013; Ekanadham et al., 2014):

$$\min_{\vec{w}} \quad \left\| \vec{s}(t) - \sum_{ij} b_{ij}\vec{w}_j(t - t_i) \right\|_2^2 \tag{2.3}$$

The two methods seem to give similar results[1] although, in theory, the first approach is less sensitive to noise, whereas, the second one is less sensitive to strong correlations between cells (i.e. overlapping spikes). This is due to the fact that taking the median is a way to minimize the $\ell 1$-norm between the different snippets and the template, while equation 2.3 is a minimization of a $\ell 2$-norm.

2.4.2 Finding the spike trains

Once the templates are found, we need to find when they appear on the extracellular signal. For this, template matching methods usually use algorithms similar to projection pursuit (Friedman and Tukey, 1974), although with different criteria for acceptance and stop). Most of them can be summarized as an iterative greedy approach with the following steps, for a given time chunk (illustrated in figure 2.1A):

1 Find the template that matches best the raw data. If amplitude is allowed to be different from 1, find the best matching amplitude.

[1] See http://phy.cortexlab.net/data/sortingComparison/ for a direct comparison on some synthetic ground-truth datasets

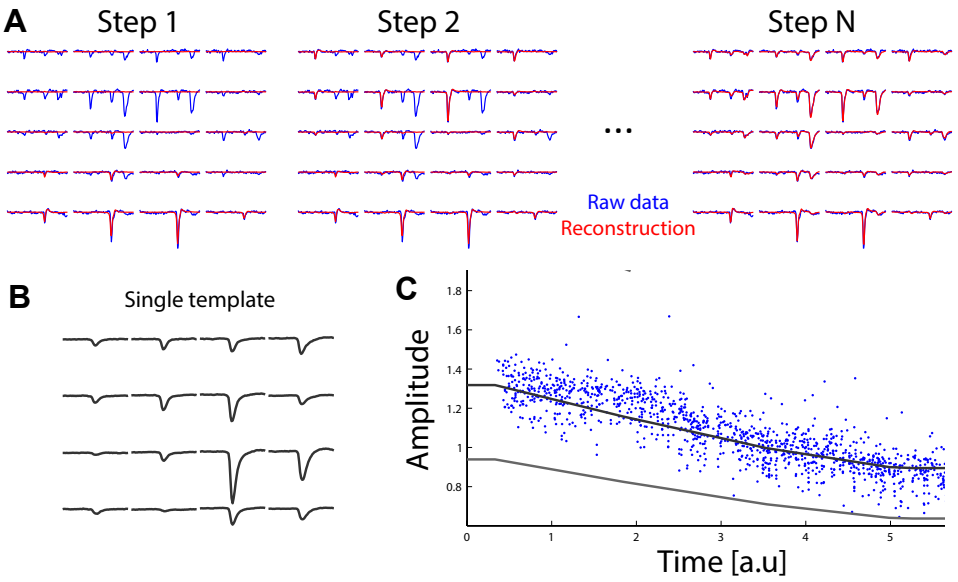

Figure 2.1: The template matching approach. **A.** Illustration of the iterative template matching approach. The extracellular signal (in blue, shown for 20 electrodes) is matched iteratively with a sum of templates. At each step, a template is added to the signal (red) to match better the data. At the end, all the spikes are fitted by a template, and the sum of templates (red) predict very well the data (blue). **B.** Example of a single template over 16 electrodes. **C.** Example of amplitude values fitted to the data for one template, as a function of time. Gray lines represent the average amplitude over time, and the minimal amplitude over time (see text for details).

2 Define a criterion to accept the template. It can either be about the quality of the fit to the raw data, or about the value of the best amplitude, or both.

3 If the template is accepted, subtract it from the raw data. Then go back to the first step.

The different algorithms that have been proposed differ mostly in the acceptance criterion, and in the possibility to have amplitude different from 0 and 1 or not.

One common issue that needs to be mentioned before comparing the approaches is sampling jitter. When a cell emits a spike, the spike time may peak at a time $t + dt$, where t is the closest time point sampled by the data acquisition, and dt is the time difference between the true spike time

and t, smaller than the acquisition period. As a result, in template matching approaches, a template will be matched at time t to explain a spike that occurred at $t+dt$. The compensation of this dt is necessary (McGill and Dorfman, 1984) when one does not use a high sampling frequency. For example, Prentice et al. (2011) use linear interpolations, Pillow, Shlens, Chichilnisky, et al. (2013) use local approximations based on Taylor expansions and Yger et al. (2016) use similar expansions (see also (Marre et al., 2012) where this issue is mentioned). Additional solutions, such as polar expansions, were developed by Ekanadham et al. (2011).

2.4.3 Approaches with binary amplitudes

Segev et al. (2004), Pillow, Shlens, Chichilnisky, et al. (2013) and Franke, Quiroga, et al. (2015) assume that the amplitude of a template is always equal to 1 ($a_{ij} \in \{0, 1\}$ in equation 2.1). Segev et al. (2004) keep a template if it improved the prediction of the extracellular signal by the sum of templates, i.e. if subtracting it to the raw data led to a reduction in variability that passes a given threshold. This threshold is needed to avoid overfitting the noise with small templates. Pillow, Shlens, Chichilnisky, et al. (2013) base the criterion of acceptance on an objective function: the value of the function had to be improved when fitting an additional spike. This function is the sum of two terms:

$$\left\| \vec{s}(t) - \sum_{ij} a_{ij} \vec{w}_j (t - t_i) \right\|_2^2 - \sum_j \gamma_j \sum_i a_{ij} \tag{2.4}$$

The first one is the square difference between the extracellular signal and the sum of templates, in the metric defined by the noise covariance. It will usually decrease if an additional spike is fitted to the signal. The second term is a regularization on the average firing rate of each cell, and corresponds to a cost per spike. This term decreases when an additional spike is fitted to the signal, and reflects the prior that cells are more likely to be silent (i.e. respect their firing rate) than to fire all the time. This second term is here to avoid overfitting the noise with small templates. Note that, while this term is called a prior by Pillow, Shlens, Chichilnisky, et al. (2013), it is based on the data (on the measured firing rate for each cell). We will call it a regularization term in the following. Conceptually, we can see that the two methods are quite similar. If we whiten the extracellular signal before template matching, then the first term in the objective function of Pillow, Shlens, Chichilnisky, et al. (2013) is equivalent to the square difference between the extracellular signal and the sum of templates, which is exactly what Segev et al. (2004) use. When Segev et al. (2004) then compare the reduction of this square difference

to a threshold, this threshold can be compared to the change of the second term in the objective function of Pillow, Shlens, Chichilnisky, et al. (2013), which reflects the regularization on the firing rate. The method of Pillow, Shlens, Chichilnisky, et al., 2013 is more elaborate because the regularization term can change from one cell to the other, while the method of Segev et al. (2004) uses the same threshold for all cells. However, it seems that the exact regularization values does not change much the results of the spike sorting (Pillow, Shlens, Chichilnisky, et al., 2013). Therefore, we expect that these algorithms should give similar results. More recently, Franke, Quiroga, et al. (2015) used a relatively similar approach but allowed fitting two templates at the same time. This additional feature leads to a better estimation in the case of overlapping spikes.

2.4.4 Approaches with graded amplitudes

Other methods have assumed that a template can be scaled up or down every time the cell spikes: they assume that the amplitude a_{ij} can take other values than 0 or 1 in equation 2.1. Prentice et al. (2011) assume that the spike amplitude for a given cell follows a Gaussian probability distribution, whose mean is equal to 1. The standard deviation of the distribution is estimated from the previously found cluster. Then, they maximized an objective function that has two terms: the first one is the same as the one of Pillow, Shlens, Chichilnisky, et al. (2013), i.e. the difference between extracellular signal and the sum of templates in the noise covariance metric. The second one is a regularization term that reflects two facts. First, an amplitude closer to 1 is more likely than a very small, or a very big one. Second, a template with a high firing rate is more likely than another one with a low firing rate. By balancing these two terms, the optimization process avoid to add a lot of templates with small amplitudes that are highly unlikely. It also avoid to add a lot of templates associated to units with low firing rates. This second term can thus be understood as a combination of two regularization constraints: one over the amplitudes, and another one over the firing rates.

Marre et al. (2012) and Yger et al. (2016) also allow amplitude variations, but the acceptance criterion was different: after having found the amplitude that best matches the extracellular signal, the template was kept if the amplitude was between thresholds, a_{min} and a_{max}. At first sight, this criterion seems surprising since it does not depend on the improvement in the quality of the fit. In fact, the process of finding the best amplitude is by itself an estimation of the improvement of the fit. For a given iteration, if we note $\vec{r}(t)$ the extracellular signal that remains to be fitted (i.e. after subtraction of the templates fitted in the previous iterations), and $\vec{w}(t)$ the new candidate

template that needs to be fitted, then the best matching amplitude a will be found by minimizing $(\vec{r}(t) - a\vec{w}(t))^2$. It can be shown that, if this template is accepted, the square difference will decrease by $a^2(\vec{w}(t))^2$. In other algorithms, this decrease of the square difference has to be larger than the increase of the regularization term for the template to be accepted. Setting a minimal amplitude is therefore equivalent to having a regularization term that is different for each cell, but constant as a function of the amplitude of the spike (similar to what was done by Pillow, Shlens, Chichilnisky, et al. (2013)). The other threshold for maximal amplitude is less important, and only plays a role to avoid very high, unrealistic values.

The advantage of having an amplitude threshold as a parameter, instead of a threshold for improvement in the goodness of fit, is that this parameter is much more intuitive for the user: we can figure out reasonably well what a minimal amplitude of 0.4 or 0.8 means. Thresholds on goodness of fit are less easy to understand. Furthermore, by looking at the set of amplitudes fitted over time, we can get a sense of the right values for these amplitude thresholds. If the minimal amplitude threshold is too low, the template is also fitted on noise, with small amplitudes which are clearly different from the amplitude of real spikes, that are close to 1. When we labeled the pairs of spikes with refractory period violations, we often see that most of them involve one of these spurious fits. It is therefore easy to readjust the threshold to a correct value. These thresholds can also be made time dependent, as can be seen in figure 2.1C. This gives more flexibility to process non-stationary data while keeping understandable parameters. Of course, the disadvantage of this method is that the algorithm is not expressed as the minimization of a cost function.

2.4.5 Different algorithms correspond to different assumptions about spike amplitude distributions

Can all these methods be expressed with an objective function having a similar structure? As we showed before, all the three methods discussed above (Prentice et al., 2011; Marre et al., 2012; Pillow, Shlens, Chichilnisky, et al., 2013) aim at minimizing the square difference between the extracellular signal and the sum of fitted templates. The difference lies in the regularization term, which reflects an assumption about the possible amplitude for the spike. More formally, the quantity we want to minimize is:

$$\min_{a_{ij}} \quad \left\| \vec{s}(t) - \sum_{ij} a_{ij} \vec{w}_j (t - t_i) \right\|_2^2 - \lambda \, \mathcal{R}\Big(\{a_{ij}\}\Big) \tag{2.5}$$

where the first term is the square difference between the data and the reconstruction model, $\mathcal{R}$ denotes the regularization function over the amplitude values a_{ij} and λ is a free parameter (i.e. trade-off between the two terms).

In (Prentice et al., 2011), the regularization term reflects an assumed Gaussian distribution for the amplitude. In (Marre et al., 2012), the amplitude thresholds might reflect an assumption of flat amplitude distribution between the minimal and maximal amplitudes, and 0 elsewhere.

With a similar approach, Ekanadham et al. (2014) use a $\ell 1$-minimization algorithm to find the right amplitudes. This $\ell 1$-minimization is equivalent to assuming that the spike amplitude distribution that has the form of a power law, i.e. $1/(\epsilon + a)^p$. The form of this distribution gives an advantage to small amplitudes. As a consequence, this algorithm outputs a lot of small amplitude spikes, and this is later corrected by removing all the spikes whose amplitude is smaller than a given threshold. The threshold is estimated *a posteriori* by fitting a Gaussian distribution to the amplitude distribution found empirically.

One way to summarize the difference between these three methods is therefore to say that they differ in their assumption on the amplitude distribution. Prentice et al. (2011) assume a Gaussian distribution, Marre et al. (2012) and Yger et al. (2016) assume a flat distribution between some thresholds, Ekanadham et al. (2014) and Pachitariu et al. (2016) assumed a power-law distribution in the core of the algorithm, but corrected it later on with a Gaussian distribution.

2.4.6 Caveats when minimizing an objective function

While this is an intuitive way to explain the differences between the different algorithms, it has to be noted that some sorting algorithms do not directly minimize the objective function described above. For example, in both (Prentice et al., 2011) and (Marre et al., 2012), during the iterative process, the amplitude was chosen as the one that best matches the data, without taking into account the regularization term on the amplitude values. Formally, the amplitudes were chosen to be the solution of:

$$\min_{a_{ij}} \quad \left\| \vec{s}(t) - \sum_{ij} a_{ij} \vec{w}_j (t - t_i) \right\|_2^2 \tag{2.6}$$

A direct minimization of the total objective function, including the regularization term, would have biased all the amplitudes towards 1 since the amplitudes would have been the solution of:

$$\min_{a_{ij}} \quad \left\| \vec{s}(t) - \sum_{ij} a_{ij} \vec{w}_j (t - t_i) \right\|_2^2 - \lambda \sum_{ij} \log \Big(p(a_{ij}) \Big) \tag{2.7}$$

where λ is a free parameter and $p(a_{ij})$ is the probability density function of the amplitude values. This bias affects the quality of the fit, and can lead to fitting other templates where templates have been fitted with biased amplitudes. For example, figure 2.2A shows a comparison of several errors function used while optimizing the amplitude of a given waveform, displayed in figure 2.2B. As we can see, the choice of the error criteria can have a strong impact on the "optimal" amplitude, leading to more or less pronounced residuals (see figure 2.2C). To avoid this, it is necessary to take the amplitude value that best matches the data, without any regularization, and only use the regularization to decide afterwards whether this template should be kept or not.

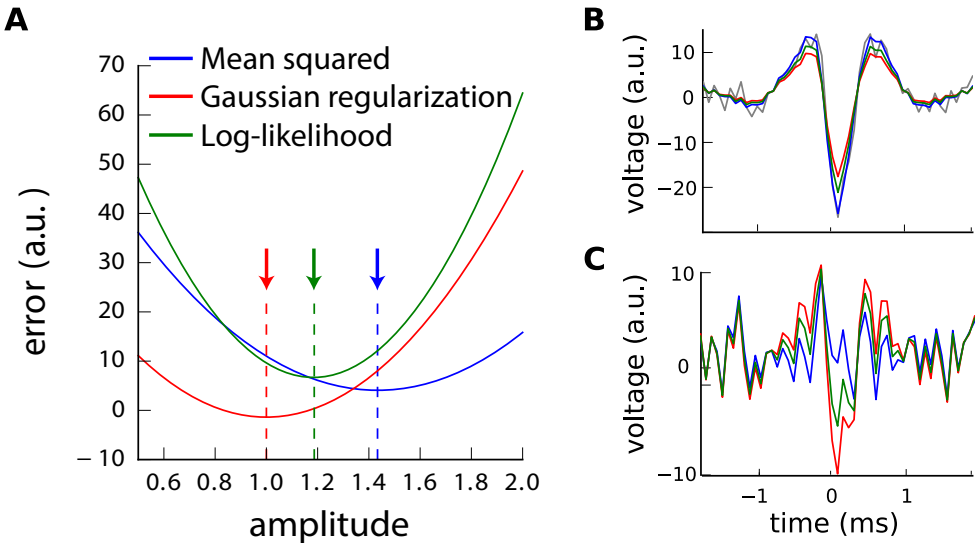

Figure 2.2: Illustration of biased amplitudes toward 1 when minimizing the log-likelihood. **A.** Comparison of the error function used for the optimization of the amplitudes. Mean squared error of the residual, as described in equation 2.6 (blue). Penalty which comes from a regularization with a Gaussian distribution on the amplitude values (red). Log-likelihood, as described in equation 2.7 (green). The dotted vertical lines indicate the minimum of each of these error functions. **B.** Illustration of the results of the fit, with optimal scaled waveforms for each error function superimposed onto the raw data (gray), color-coded as in **A**. **C.** Residuals (fit minus raw data) for each of those error functions, color-coded as in **A**.

2.4.7 Assumptions behind the template decomposition

An important question is whether template matching algorithms can always replace clustering algorithms, or if they have some intrinsic limitations that make them less flexible than clustering. This is still an open question, and only direct comparisons between the different approaches, in cases where the true solution is known, will tell us what is the best approach (Yger et al., 2016). Here we would like to give some intuition about how the main assumptions of template matching approaches can translate in the feature space that clustering approaches use.

In the template matching approach, the noise is supposed to be independent of the templates. In the case where no amplitude variation is allowed, it means that the variability in the snippets always comes from the same noise source. In a given feature space, it means that all the clusters should be elongated in the same directions. This is illustrated in figure 2.3A: while the clusters have different centers, they are all ellipses extended in the same directions.

If the spike amplitude is allowed to change, this means that, in a feature space, each cluster has two sources of variability: a common one, which corresponds to the noise, and another one that is specific to each template. The second one is constrained to be in the direction of the template, which is approximately the cluster center. Therefore, in a feature space, it means that the clusters have now noise in common directions, but also an elongation that will follow the arrow that connects the point 0 in the feature space, and the center of the cluster (figure 2.3B). This is more realistic than the previous assumption, but it is not clear whether this gives a good account of all the variability found for each cluster.

If we were to use template matching only on isolated spikes, we could also define areas in the feature space where a point is assigned to a given template. A snippet is always assigned to the best matching template. In some algorithms (Pillow, Shlens, Chichilnisky, et al., 2013), it means this template is closest in the sense of the least square difference. In the feature space, this means that a point will always be assigned to the closest centroid. We can use this rule to define equivalent cluster borders (figure 2.3C). In other algorithms (Prentice et al., 2011; Marre et al., 2012), only the spike shape is used to define the best matching template, and then the algorithm decides whether the best matching amplitude is plausible or not. This defines different shapes for the border: a straight line from the 0 point to separate the regions of preferred spike shape and some circles to define the allowed amplitudes, following the approach of Marre et al., 2012. Figure 2.3D illustrates these shapes.

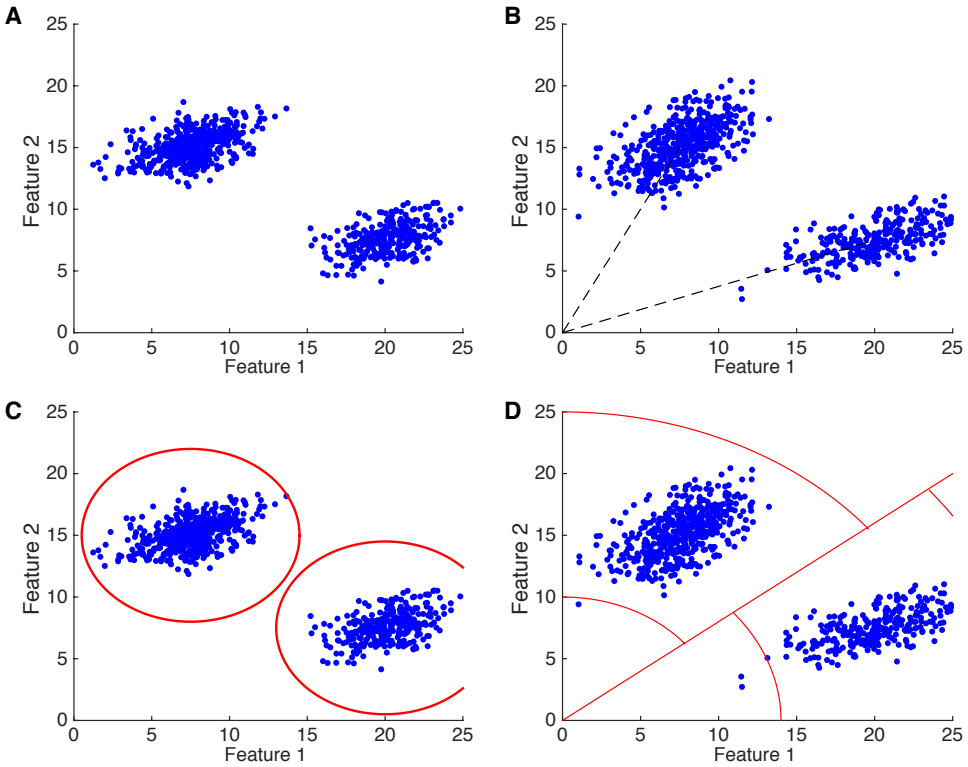

Figure 2.3: Illustrations of the assumptions of the template matching in the clustering space. **A.** Example of two clusters in the feature space, when assuming that they are generated by templates with no amplitude variation. **B.** Same than **A**, but now with the assumption that the template can vary in amplitude according a Gaussian distribution. **C.** Equivalent borders (see text) for the clusters for a template matching that chooses the template closest to the spike. **D.** Equivalent borders in the case where the template is chosen based on the spike shape, and that only a certain range of amplitude is allowed. See text for details.

So the competition between the different templates defines some natural borders. There is no guarantee that this is the best and proper definition for the cluster borders. Future works will need to address this issue by comparing the results of template matching and clustering algorithms. However, the intuitions we have drawn here can be used to compare more intuitively pure clustering-based versus template matching approaches.

2.5 Conclusion: challenges ahead

The methods described here have enabled to sort spikes from a large number of cells and electrodes (Yger et al., 2016; Pachitariu et al., 2016). However, there are still several challenges that need to be overcome. First, most of the algorithms described here have been tested on *in vitro* data, in the retina (but see (Ekanadham et al., 2014; Franke, Quiroga, et al., 2015; Yger et al., 2016) for *in vivo* tests). *In vivo* tests on silicon probes with a large number of recording sites close apart will be necessary. A possible required improvement is a better description of the cluster (Yger et al., 2016). As we explained above, the template matching makes some assumptions about the shape of the clusters, and it is not clear if these assumptions are verified or not *in vivo*. A related issue with spike sorting is the need to have more ground truth data, i.e. recordings where at least one cell is recorded with another technique, so that we know when the spikes occur. These data are essential to test spike sorting algorithms (Neto et al., 2016).

A second point is that template matching does not replace clustering. All the methods described require a set of clusters, from which the templates can be extracted. The clustering can do mistakes that can be tolerated, as long as they do not distort the template estimation. But a decent performance in clustering is nonetheless required. So one still needs an efficient way to cluster. Ekanadham et al. (2014) and Pillow, Shlens, Chichilnisky, et al. (2013) have proposed to do back and forth between template estimation and finding the amplitudes. This is an extension of the approach we described previously: after finding the amplitudes, they are used to estimate the templates again with a least square method. Then this new set of templates is fitted once again to the data. Note that this global iteration does not remove the need for an initial clustering, so that the templates are properly initiated (at the very least, they need to be in sufficient numbers). The interest of doing multiple iterations of template estimation and matching is not completely clear. While Ekanadham et al. (2014) claim that it is crucial, Pillow, Shlens, Chichilnisky, et al. (2013) mention that there is only a marginal improvement after the first pass. Another modification of the iterative approach can be found in a work of Franke, Quiroga, et al. (2015), where solutions beyond this iterative approach have been developed that can lead to a better sorting of synchronous spikes.

Another challenge is the time spent on manual curation. Even the best clustering makes mistakes, and some cells will be represented by more than one template. Finding all the pairs that need to be merged require a significant amount of time for hundreds of electrodes. Methods need to be developed to make this kind of tasks as automated as possible, so that the

time spent by the user is reduced to a minimum (Yger et al., 2016).

Nowadays, new devices with CMOS components now allow recordings from thousands electrodes simultaneously (Berdondini et al., 2005; Fiscella et al., 2012; Müller et al., 2015; Hilgen et al., 2016), and it remains to be seen it these algorithms can scale up and process such a large amount of data. We need to be sure that the time spent on manual curation can remain small enough that we can get thousands of spike trains in a decent amount of time (see preliminary evidence that it might be the case by Yger et al. (2016)).

Finally, one problem that needs to be properly tackled by the new generation of spike sorting algorithms appears during long lasting chronic recordings (Nicolelis et al., 2003). It is indeed well known that because of tissue changes, or because of experimental protocols, recordings can be non-stationary and drifts in the neuronal waveforms can appear over long time scales. For any template matching based approach, one should rather consider spatio-temporal kernels that could evolve over time, and be distorted. To some extent, some of these deformations can be dealt with by allowing graded amplitudes for the templates (see for example figure 2.1C, where the amplitude evolves over time). However, a more robust framework is required for a better understanding of the drifts, especially because latest algorithms (Yger et al., 2016; Pachitariu et al., 2016) seem to pave the way toward real-time spike sorting. Such an understanding would be crucial in the context of accurate online spike sorting.

2.6 Acknowledgments

This work was supported by ANR OPTIMA and TRAJECTORY, the French State program Investissements d'Avenir managed by the Agence Nationale de la Recherche [LIFESENSES: ANR-10-LABX-65], a grant from the European Union Seventh Framework Programme (FP7/2007-2013, grant agreement no. 604102, Human Brain Project), and NIH grant U01NS09050 to OM.

Chapter 3

Online spike sorting

We have seen that efficient algorithms have been developed to estimate the individual spiking activity of multiple neurons from voltage traces recorded with dense and large-scale microelectrode arrays. They are able to reach good performances both on synthetic and real (*in vitro* and *in vivo*) ground-truth recordings. While the associated software are now available as standardized toolboxes, it is worthwhile noticing that all of them are meant to analyze the data offline, i.e. after they have been recorded experimentally.

While spike sorting is a computationally expensive task, parallel and distributed computing allow analyzing recorded data in a time comparable to the recording duration. This suggests that they could be analyzed online, i.e. during the time course of the experiment. Online spike sorting would open important possibilities to better probe dynamically the responses of large-scale neuronal populations with closed-loop experiments.

Here I present a new generic toolbox for online spike sorting which allows resolving the spike trains of each recorded cells in firm real-time, i.e. infrequent deadline misses are tolerable. This toolbox is an extension of an already well-established and highly used offline algorithm, called SpyKING CIRCUS (Yger et al., 2018) to which I contributed. I present a modular and distributed architecture which is necessary to split the task into multiple processes and handle synchronous communications between them. Finally, I validate the performance accuracy of this software with both synthetic data and real ground-truth recordings.

This chapter corresponds to an article in preparation in collaboration with Pierre Yger and Olivier Marre.

3.1 Introduction

In recent years, multi-electrode arrays and large silicon probes have been developed to record simultaneously from hundreds of electrodes packed with a high density, both *in vitro* (Berdondini et al., 2005; Fiscella et al., 2012; Lambacher et al., 2004) and *in vivo* (James J. Jun et al., 2017). Such recordings should allow neuroscientists to reconstruct, with the help of signal processing algorithms, the individual activity of many neurons spiking in the vicinity of those electrodes, and thus have access to the simultaneous activity of hundreds of cells, at the single spike resolution. However, obtaining this information is a very complex problem of source separation, also known as "spike sorting" (see (Lefebvre et al., 2016) for a review on the subject). Driven by the constantly increasing density of the electrodes, newly developed algorithms (Pachitariu et al., 2016; James J Jun et al., 2017; Lee et al., 2020; Yger et al., 2018; Hilgen et al., 2016) have focused on the need of reproducibility, validation, speed and automation (Magland et al., 2020; Buccino et al., 2019). Nowadays, many of these software are available as standardized toolboxes with various pros and cons, depending on the implementation details that can strongly differ from one to the other. However, it is worthwhile noticing that all of them are meant to work offline, i.e. to be launched after the data have been acquired.

Online, automated and scalable spike sorting would open important possibilities to better probe dynamically the responses of large-scale neuronal populations. Closing the loop between the recorded neurons and the external stimulation, allowing modulation of the experiment on-the-fly as a function of the recorded responses, would open new types of experiments (Benda et al., 2007; Bölinger and Gollisch, 2012; Gollisch and Herz, 2012). These closed-loop paradigms can be very useful, for example for neurofeedback (Sitaram et al., 2017). So far, very few attempts have been made for online spike sorting (Rutishauser et al., 2006; Franke, Natora, et al., 2010; Nguyen et al., 2014). In most cases, online sorting was simply achieved by deriving simple decision rules based on the spike templates (Guenther et al., 2009), but none of them was applied in the context of high-density electrophysiology. While online spike sorting would bring much more information than naïve estimation of the average firing rate by voltage thresholding on a single electrode (for example when decoding visual inputs from multiple neurons in the human temporal lobe (Quiroga, Reddy, et al., 2007)), most of the brain machine interfaces still rely on the latter, because of its simplicity (Christie et al., 2015). Note that some authors have shown that voltage thresholding can be sufficient for specific applications such as neuroprosthetic control, thanks to the rather low dimensional neural activity (Fraser et al.,

2009; Todorova et al., 2014; Trautmann et al., 2019).

The density of modern probes advocated for a change in the classical pipeline of spike sorting algorithms (Lefebvre et al., 2016). Because channels are recording the activity of many cells, the amount of so-called spatiotemporal overlaps between simultaneously firing neurons is now rather large in the context of high-density probes. Therefore, all algorithmic approaches solely based on clustering (Lefebvre et al., 2016; Einevoll et al., 2012) cannot be applied online, or at the cost of very poor performance. This is for example the case of fuzzy C-mean clustering (Oliynyk et al., 2012), or density stream clustering (Rutishauser et al., 2006)). Since these algorithms can not solve the problem of overlapping spikes, to avoid this issue, other solutions have been proposed, such as the use of linear filters (Franke, Natora, et al., 2010; K. H. Kim and S. J. Kim, 2000; Wouters et al., 2018). However, none of them has yet been properly tested and validated in the context of high-density recordings.

Here we present a generic toolbox for online spike sorting of both *in vitro* and *in vivo* recordings. It is an extension of an already well-established and highly used offline algorithm, called SpyKING CIRCUS (Yger et al., 2018), based on a combination of density-based clustering and greedy template matching. In order to achieve online accurate spike sorting in the context of high density electrophysiology, we designed a modular and distributed Python architecture to handle synchronous communication among processes, identified as key computational step of our offline algorithm. We then validated our online software first with synthetic data, before comparing its performance with the one obtained by its offline counterpart, on real ground-truth recordings (Spampinato et al., 2018).

3.2 Results

3.2.1 A distributed architecture

To better understand what the current problems with online spike sorting are, one should note that most of the existing accurate approaches for offline spike sorting (Pachitariu et al., 2016; James J Jun et al., 2017; Lee et al., 2020; Yger et al., 2018) can be decomposed conceptually in two main steps (see figure 3.1A). First, assuming data have been appropriately pre-processed (mostly high pass filtered and whitened), a subset of the spikes is projected in a low-dimensional space and characterized by a small set of numbers/features. In such a reduced space, a clustering algorithm is launched on this subset in order to identify the centers of the clusters. These centroids are the so-called

"templates", i.e. spatiotemporal motifs characterizing the effect of a single spike on the nearby extracellular recordings. Second, once this dictionary of templates has been found, the templates can be matched to the raw data in an iterative and greedy manner in order to resolve the spatiotemporal overlaps (Yger et al., 2018).

Some algorithms optimize both the spike times and the templates alternatively (Pachitariu et al., 2016), and thus need to do iterative passes over the data before convergence. However, in the case of SpyKING CIRCUS, if we assume that the dictionary of templates is available (figure 3.1A), assigning labels to newly incoming spikes can be performed online and on-the-fly. This property motivated all the previous approaches trying to implement the templates as linear filters on Field Programmable Gate Arrays (FPGAs) (Franke, Natora, et al., 2010; K. H. Kim and S. J. Kim, 2000; Wouters et al., 2018). However, in this paper, we decided to tackle the problem from a different angle, and adapt a well-established toolbox for spike sorting (SpyKING CIRCUS, see (Yger et al., 2018)) in order to work online, with data buffers. This assumes that the fitting procedure is fast enough to handle the stream of newly incoming data buffers. To this aim, we identified the canonical computations required by such a template-matching pipeline figure 3.1B. While adapting the SpyKING CIRCUS algorithm was our primary goal, we wanted to be as generic and flexible as possible, because there is no consensus on what is the best workflow to perform spike sorting.

After we identified the computational blocks of our online spike sorting pipeline (see figure 3.1B), we implemented a modular Python library to easily deploy and configure such a pipeline (see Material and methods). By doing so, we created a distributed architecture able to turn every computation flagged in figure 3.1B into a standalone process, receiving some input data, and sending results. Since the code has to be scalable and work in an online context, we decided to use a Remote Procedure Call (RPC) architecture (Ben-Ari, 2006) in order to share the load among several machines and control all processes from a single master node. More precisely (figure 3.1C), a `manager` can spawn Python processes P_i and distribute them over a group of machines, depending on the load/architecture. Each process implements what we are calling a `block`, i.e. a Python object able to receive and send messages, either synchronously or asynchronously. The communications among the blocks are built on top of the ZeroMQ library (Hintjens, 2010), ensuring fast and efficient transmission of the messages. The `manager` is able to control all the remote processes, and is in charge of establishing the connection among them before the start of the algorithm.

All the blocks implemented by default are the ones needed to reproduce the behavior of the SpyKING CIRCUS software (Yger et al., 2018) (see an

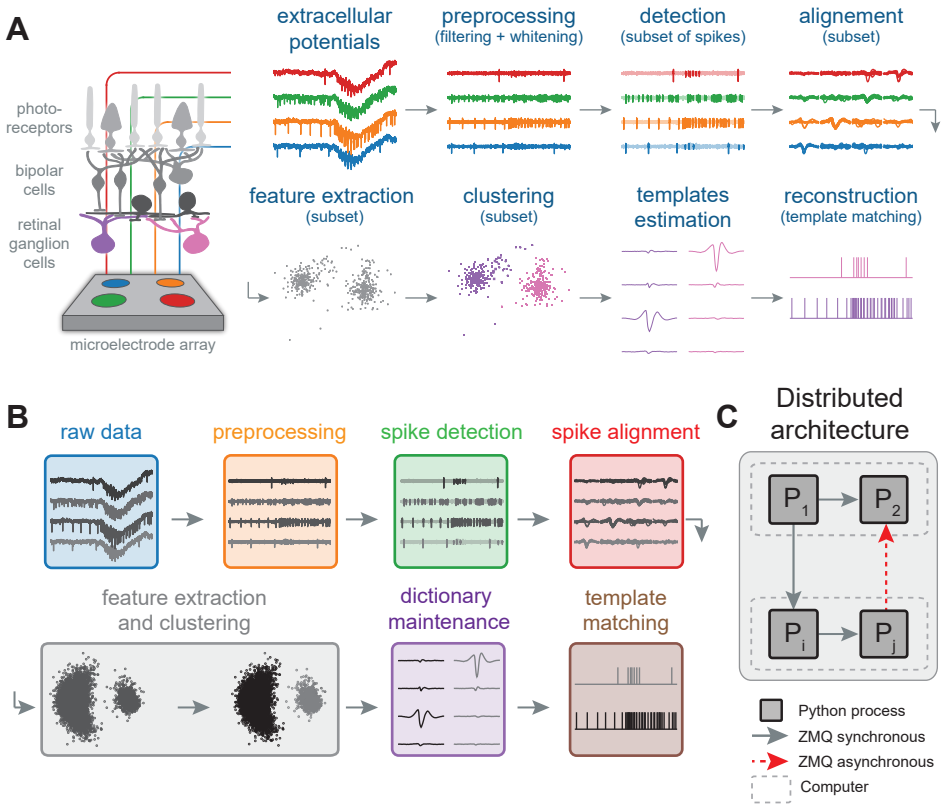

Figure 3.1: Main computational steps of template matching based spike sorting algorithm. **A.** Illustration of the canonical workflow of spike sorting algorithms based on template matching. Extracellular signals are recorded by a dense electrode, and then high-pass filtered to remove the Local Field Potentials, and eventually whitened to remove spurious spatial correlations. Then, a subset of all spikes is detected, aligned (possibly with up-sampling), and projected into a lower dimensional space by feature extraction. A clustering algorithm identifies the centroids, i.e. the templates, and constructs the dictionary. Given the dictionary of templates, the signal is reconstructed as a linear sum by greedy template matching methods, to obtain the spike trains. **B.** The key computational steps are isolated in order to be implemented in a modular manner. **C.** Schematic of the modular architecture used to implement online spike sorting. Python processes can be distributed over several machines, and communicate via ZMQ buffers.

exhaustive list in table 3.1). To give a brief overview of the full pipeline, illustrated in figure 3.2A, a preprocessing step (involving filtering the raw data, then computing a whitening matrix) is followed by a spike detection step. To detect the spikes, we need to estimate the median absolute deviations θ_i for all channels (as running averages over time, see Material and methods), and use these values to get the threshold crossings over all channels, called "peak times". The peak times are then used during a clustering step, in the following manner. First, snippets of data around these events are collected on single channels, simply to get a picture of what typical waveforms look like. The snippets are up-sampled and realigned (see Material and methods), and once enough have been found, a Principal Component Analysis (PCA) is performed to reduce the dimensionality (see Material and methods). Once the PCA matrix has been defined, aligned and reduced spatiotemporal snippets are collected, pooled with respect to the channel where they show the highest peak (as a divide and conquer approach, see (Yger et al., 2018)), and independent density-based clusterings are launched when enough snippets are collected for each pool/channel (figure 3.2A). As soon as the independent clusterings are finished, templates (defined as the centroids of the clusters) are sent to a template updater node, in charge of keeping the dictionary of templates up-to-date, and discarding any possible duplicates. This node works asynchronously, receiving templates on-the-fly when they are sent by the clustering processes. Finally, the template matching step is combining the peaks, the filtered data and the templates in order to reconstruct the signal as a linear sum of templates, and for each of the fitted spikes, outputs a template identifier and an amplitude coefficient (see (Yger et al., 2018)).

Since the code is meant to work online, it has to deal with data buffers captured on-the-fly by the acquisition board. By definition, we are aiming for what's called "soft" real-time (see figure 3.2B): there is no hard limit for the time devoted to process one data buffer (and thus there could be some lag). However, on average, the code should process the data buffers on time. Note that the size of the buffers can be customized or constrained by the manufacturer, but in the following we always worked with incoming data from a multi-channel acquisition board, setting 1024 data points per buffer, at a sampling rate of 20 kHz (see Material and methods). Since the duration of the buffer is an intrinsic limit for the lag of the system, in our case it means that the minimal lag expected is around 50 ms. It is important to stress that such a lag may prevent the use of such a template-matching based architecture for experiment that may require very fast temporal resolution, such as for example Spike Timing Dependent Plasticity (STDP) protocols (Markram et al., 1997).

In order to ensure the scalability of the code, we also offer the possibility, for some identified nodes, to be multiplexed either in space or in time, with a number of processes k controlled by the user. Indeed, we noticed that some computation such as filtering, thresholding or even clustering (as long as the spikes are pooled by channels) could easily be multiplexed in space, i.e. several processes could be in charge of subgroups of channels (since the processing are independent) (see figure 3.2C). By doing so, we can obtain a linear speedup for these key operations. Similarly, for some operations that may be time consuming and crucial, we offer the possibility to multiplex some `block` in time, such as the peak detection or the fitters (see figure 3.2C). More specifically, it means that instead of having one process working on all data buffers, we have k processes and each of them is handling only 1 buffer out of k. By devoting more processes to the computation, we are preventing any spurious lag that could provoke successive deadline misses. This allows to avoid congestion by having processes which work on less data that they can handle, and deliver the results of the computations online as expected.

3.2.2 A scalable design

The modular design implemented with our Python processes is meant to be scalable, thus offering a flexible and generic Application Programming Interface (API) for any spike sorting pipeline that could be described with a communication flow similar to the one of figure 3.2A. However, the major bottleneck for online spike sorting in the context of dense probes is to ensure that the fitting procedure (the fitter `blocks`) is fast enough to be able to work in real-time. To check this, we first considered the situation where the dictionary of templates is known, and we were only focusing on the speed of the fitters. This is equivalent, in figure 3.2A, to remove the clustering `block` (dash-dotted region) and provide directly the dictionary of template to the fitters. Using synthetic data generated from a known set of templates (similarly to MEAREC (Buccino et al., 2019), see Material and methods), we tested our algorithm in two specific configurations, that could slow down this fitting step. First, by increasing the number of channels, from 4 to 1024, keeping the density of cell constant, to 0.25 cells per channels. Secondly, by keeping the number of channel constants at 256 and increasing the density of the cells.

To process data online, each node should process one second of data in less than one second which defines a real-time limit. As can be seen in figure 3.3A, all the nodes involved in the online procedure are operating below the real-time limit, when the number of channels is varied and the density (i.e. the number of cell per channel is kept constant). For the `blocks` that

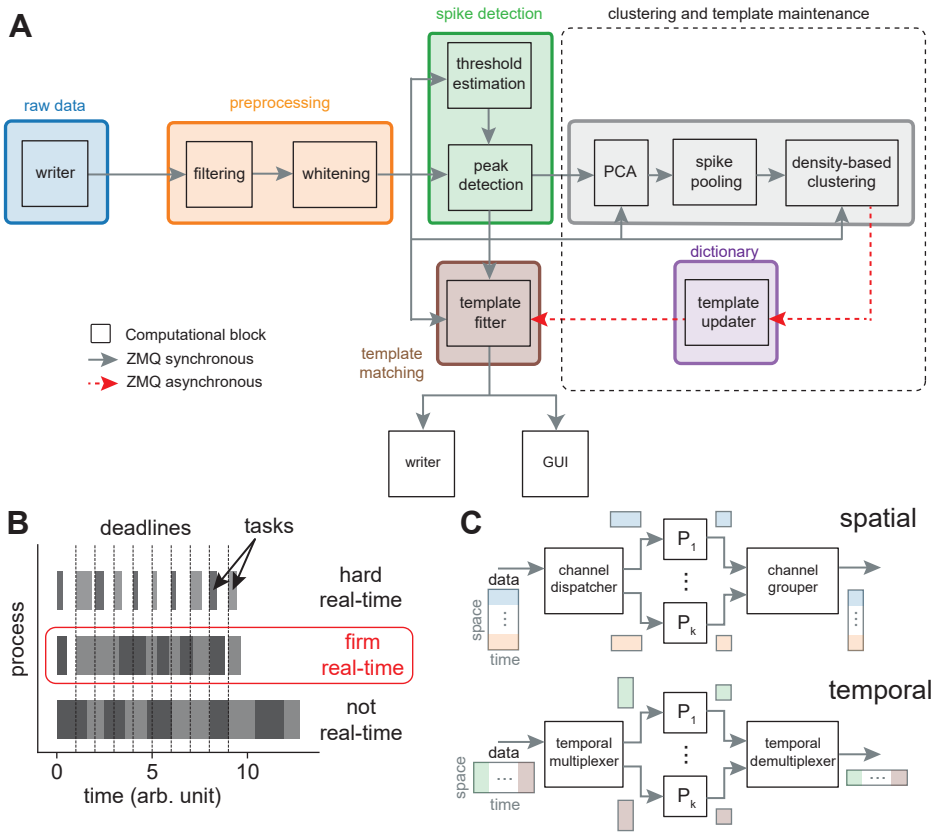

Figure 3.2: Implementation of the online pipeline. **A.** Description of all the computational `blocks` involved in the online sorting, alongside with the communications and the communication buffers, either synchronous or not. **B.** Illustration of the firm real-time mode implemented by the algorithm. Processing time is, on average, faster than real time. **C.** Illustration of the multiplexing solutions used either in space (load divided by channels groups over several processes) or in time (load divided by temporal buffers over several processes).

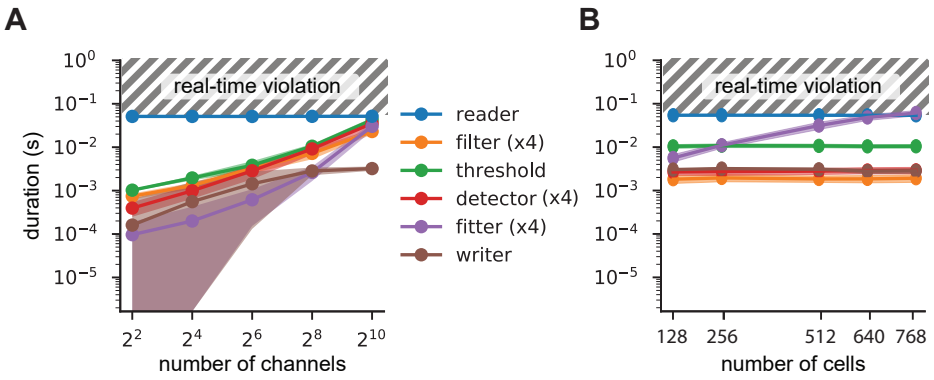

Figure 3.3: Scalability and real-time performance. **A.** Duration (in second) of all the key computational `blocks` implemented by the algorithm, as function of the number of channels. For some `blocks` (filter, detector, fitter), we used 4 processes by multiplexing either in space or time. The density of cell is kept constant, at 0.25 cells / channel. **B.** Same but when the total number of cells is varied, while the number of channels is kept constant at 256.

have been multiplexed (filtering, peak detection and fitting), we indicated the number of processes devoted, but more could be used if necessary. Combined all together, the architecture is thus able to handle 1024 channels and 256 templates online, assuming all the processes involved (19 including the reader which simulate the data acquisition) are properly distributed among machines. It is important to note that the benchmark here is also assuming "realistic" firing rates for the templates, i.e. 1 Hz. Similarly, we can see in figure 3.3B that the software is able to perform online sorting when the number of channels is kept fixed, at 256, and the cell density is varied. With 256 channels, the major bottleneck is the fitting procedure and it can work online if 4 processes are devoted to the task, with up to $\simeq$ 640 templates.

3.2.3 Validation with synthetic dataset

We benchmarked the performance of this online template-matching pipeline, similar to what have been done in (Yger et al., 2018). This pipeline assumes that the clustering procedure can be done once for all at the beginning of one experiment to acquire all the templates. We generated synthetic recordings where artificial templates are used to generate ground truth dataset, and spike times are known and controlled by the user (see Material and methods, and figure 3.4A, this is similar to what is done for example with MEAREC

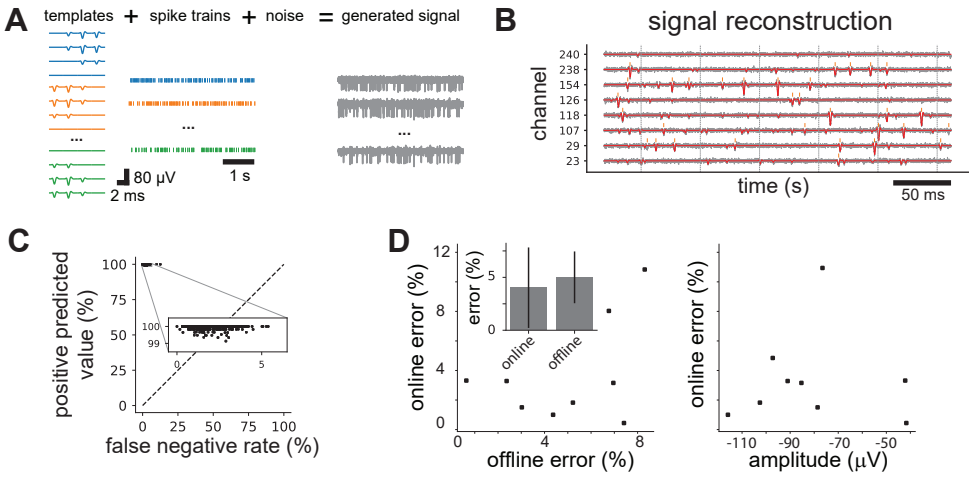

Figure 3.4: Performance of the greedy template matching procedure. **A.** Generation of synthetic dataset. Artificial templates are generated (see Material and methods), with spike times driven by inhomogeneous Poisson Processes, and then added to noise at the appropriate times. **B.** 300 ms of reconstruction (red curves) of the artificial dataset (gray curves) by the algorithm, for a subset of channels, assuming the correct dictionary of templates is provided to the template matching step. **C.** Error rates (see Material and methods) for 768 artificial templates with various noise levels. **D.** Errors obtained on the ground truth recordings by the online sorting and the offline spike sorting algorithm SpyKING CIRCUS (left). Online errors as function of the extracellular amplitudes of the spikes (right).

(Buccino et al., 2019)). Since all the computational `blocks` are very similar if not identical to the ones of the offline algorithm, we do not expect a big difference in term of quality of the sorting. Figure 3.4B shows the quality of the reconstruction, superimposing raw data (gray curves) with the reconstruction performed by the template matching algorithm (red curves). To get a more quantitative result, we computed the error (defined as the average over false negative rate and positive predicted value, see Material and methods) for 768 randomly generated templates injected with Poisson spike trains (firing rates of 1 Hz), and recorded with 256 electrodes during 10 min (sampling rate of 20 kHz). As shown in figure 3.4C, the average error rate is at around 1 %, in line with numbers obtained for the offline implementation.

3.2.4 Validation with ground truth dataset

In order to assess the performance of the software in a more realistic situation, we tested the software on ground-truth recordings already published (Spampinato et al., 2018) (see Material and methods). These are 5 min long recordings obtained with mice retina *in vitro*, and during which the activity of one retinal ganglion cells is recorded juxtacellularly, simultaneously to the extracellular recordings. Because the online pipeline might take some time to obtain the templates, every recording was looped twice, and performances were evaluated only on the second portion, where templates are considered to be acquired. The error rate is quantified as described in the Material and methods section, and in figure 3.4D, we compared the performances of the online software, presented here, with the ones of the offline sorter already published (Yger et al., 2018). As can be seen in the left panel, the performances are roughly similar for the two sorters, with a slight advantage for the online pipeline at the cost of a higher variability. In the right panel, we show that the performances do not strongly depend on the amplitudes of the spikes.

3.2.5 Drift tracking

In the previous sections, we always assumed that the dictionary of templates can be acquired once for all at the beginning of the experiment. While such an hybrid "offline/online" approach is in line with most of the solutions implemented so far for online spike sorting relying on FPGAs or linear filters (Valencia and Alimohammad, 2019; K. H. Kim and S. J. Kim, 2000; Wouters et al., 2018; Saeed and Kamboh, 2013; J. Park et al., 2017; Navajas et al., 2014), it requires strong assumptions. First, as can be seen in figure 3.5A (top), it implicitly assumes that the recording are stationary enough such that the dictionary of template can be valid during the whole course of the experiment. Second, it also assumes that the dictionary of templates acquired before the experiment is complete, i.e. that it represents the full diversity of all the cells active afterwards. In a real online experiment, this can be problematic, especially if users want to close the loop and display new stimuli as function of the population responses (see figure 3.5A bottom). These new stimuli may activate new cells, which are likely to not be present in the dictionary.

To prevent such a situation, we implemented an extension of our density-based clustering to work with data stream. In contrast with the previous "offline/online" approach, this entire online mode can collect spikes on-the-fly, and update the dictionary of templates over the course of the experiment.

For this purpose, we relied on the concept of micro clusters (Silva et al., 2013; T. Zhang et al., 1996; Aggarwal et al., 2003; Cao et al., 2006), offering a compressed representation of the cluster feature, extended into the temporal domain. More precisely, we adapted our clustering algorithm to the methods described in (Amini and Wah, 2011). After an initial clustering step, every cluster is described by a set of so-called features **CF** (see Material and methods). To summarize, the feature vector **CF** keeps track of how large the cluster is, and what is its center and extend. A micro cluster simply extends the definition of the feature vector **CF** into the temporal domain by using a fading memory, giving more weights to the data points recently added to the cluster (see Material and methods). Micro clusters can then be divided into two categories: the core micro clusters, i.e. the ones that are big enough, representing many spikes, and the outliers (few spikes). These outliers can be noise (and be pruned over time), or clusters that are building up, and eventually will become core micro clusters over time, and thus be added into the dictionary of templates. Figure 3.5B shows the representation of cores and outliers micro clusters, with the underlying data points, weighted as function of their arrival times. The later, the smaller the dots, and the less they contribute to the estimation of the micro-clusters (see Material and methods). The distinction between core and outliers is just made based on a density criteria (see Material and methods).

Regularly, when enough new spikes have been collected, we looked at the micro-clusters in order to decide if new templates have emerged, and/or if old ones are drifting and should be updated. This is illustrated in figure 3.5C, which depicts the results of the clustering at time t_0 and then later on, at time t_1, after enough new events have been collected. The software provides a module where users can specify how they want to ensure the continuity of the labels over time, i.e. what are the rules that should be used to decide if clusters at time t_1 are new clusters, or drifting ones already visible at t_0. While in practice, as in (Shan et al., 2017), we could make use of the history of the centroids to establish the matches, the default implementation is simpler. As shown in figure 3.5C, we assume that between two clusterings, performed on the core-micro clusters at two consecutive times, a cluster is "drifting" if the position of its centroid c_i at time t_1 is close to a centroid c_j that was visible at time t_0. If this is the case, then the template is updated. Otherwise, if no centroids visible at t_0 are found in the vicinity of c_i at t_1, then we assumed this is a newly formed cluster, and it is added to the dictionary of templates.

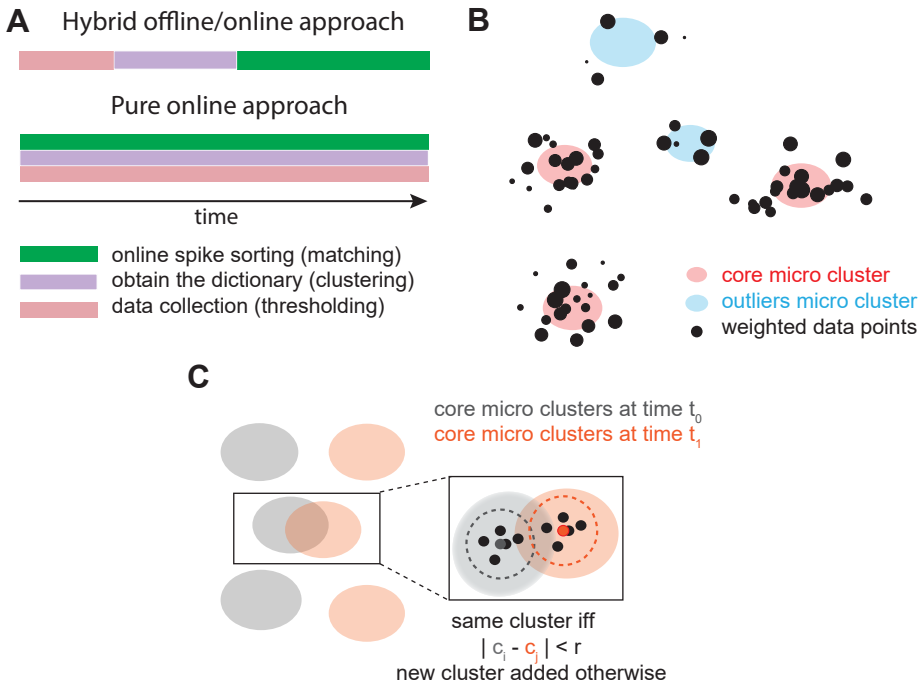

Figure 3.5: Online tracking of the drifts. **A.** Illustration of the two strategies for online spike sorting. Top: we sequentially collect data, then estimate the dictionary of templates using the collected data, before launching the fitting procedure, on-the-fly, with the static dictionary of templates obtained. Bottom: data collection, estimation of templates and fitting are all running simultaneously, such that newly collected data can modify the dictionary (updates), and be propagated at the template matching level. **B.** Compressed representation of the clusters, via the concepts of micro clusters. Every incoming spikes (black point) is either used to refine the definition of a core micro-cluster (if close enough), or kept in memory. Influence of every spike is weighted by a fading memory function (see Material and methods). Core micro-clusters (red circles) are micro clusters that are dense, while outliers micro clusters are the ones that are sparse (blue circles). **C.** Tracking of clusters.

3.2.6 Online visualization

In order to visualize the output of the algorithm, we built a modular Graphical User Interface (GUI) allowing the users to get a sense of what is happening online. Because the algorithm had to deal with large amount of data in case of thousands of channels, we decided to rely on a fast and optimize Python library called `VisPy` (Campagnola et al., 2015), similarly to `phy` (Rossant and Harris, 2013). The library is designed for high-level visualization of very large dataset, harnessing the power of the Graphical Processing Unit (GPU), using OpenGL (Shreiner et al., 2013). As can been seen in figure 3.6A, the GUI is designed as an extra `block` in our Python architecture. This `block` is meant to receive various data streams, listed in figure 3.6A, and adding new visualizations should be straightforward. While several core views are offered by the GUI, we anticipated that with online experiments, every protocol might require a specific view tailored to particular needs. Therefore, the system needs to be flexible enough to ease the implementation of new views. Figure 3.6B shows that some views are said to be "interactive", i.e. the content is dynamically adapted as function of the selection of the templates made by the user, while some are not. Figure 3.6C shows a screen shot of the online GUI, displaying some canonical properties of the fitted templates such as firing rates, spike amplitudes or distributions of inter-spike-intervals.

3.3 Discussion

Pioneer work (Nguyen et al., 2014; Gollisch and Herz, 2012) already showed how closed-loop optical neural stimulation can be coupled to extracellular recordings with tens of channels and online spike sorting. However, in addition to not being properly validated with ground truth experiments, these algorithms were not meant to be scaled up and cannot handle large numbers of electrodes. In this work, we presented a distributed architecture allowing online and accurate spike sorting for up to thousands of channels. The software is able to reach the good performances obtained by the offline algorithm SpyKING CIRCUS (Yger et al., 2018), both on synthetic and *in vitro* ground-truth recordings. It thus offers the possibility to perform real closed-loop experiments, assuming a lag of at most 100 ms is acceptable. Results can be visualized online in a modular Graphical User Interface, and used to modify dynamically the stimulus displayed during the course of the experiment.

In a simple use-case, the dictionary of templates (units) is computed after an initial period of activity controlled by the experimentalist, and then spikes

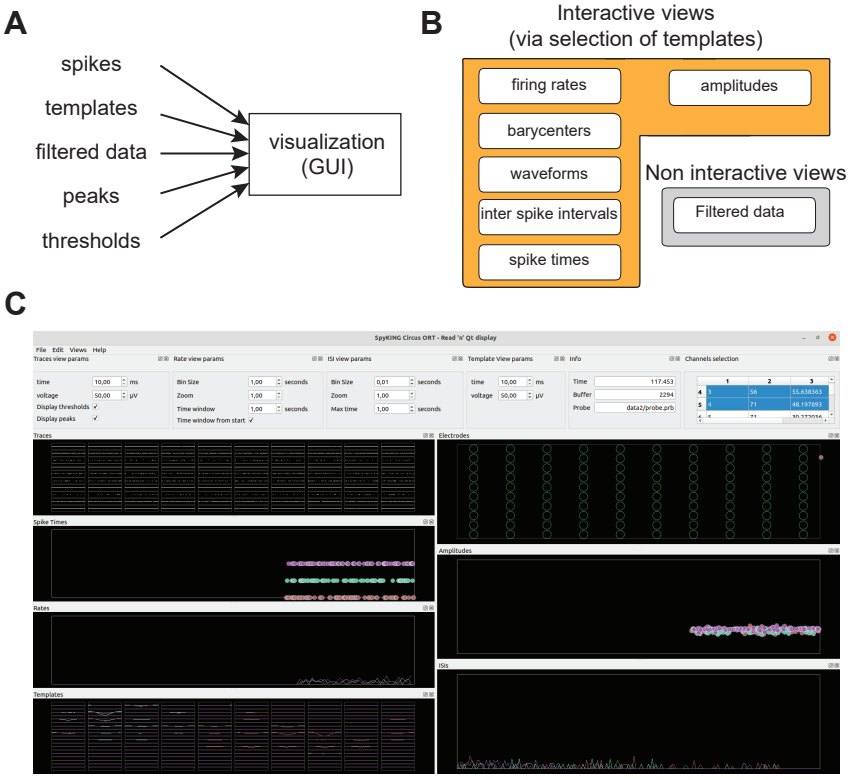

Figure 3.6: Online visualization of the spike sorting results. **A.** A visualization `block` receives various heterogeneous data buffers, in order to display meaningful information to the user, in real-time. **B.** Some views are interactive, i.e. the user can select a subset of templates and thus control what is displayed, while some other are not affected by this selection. **C.** Screenshot of the online Graphical User Interface (GUI), where users can customize the views for their needs.

are labeled on-the-fly given this static dictionary. Using greedy template-matching, the dense extra-cellular signals can thus be reconstructed online, and spikes be available for online analysis. However, since we anticipated that such a simple use-case may not be robust enough to cope with non-stationary drifts and/or complicated close-loop protocols, we implemented our clustering algorithm in a way that it can copes with data streams, and modify/create new templates during the course of the experiment. This is achieved via an hybrid "online/offline" approach, where newly incoming spikes are used to either refine the already available centroids, or kept in memory. While enough similar spikes have been gathered, a new template is created and added into the dictionary.

The effective tracking of individual units in presence of drifts is still a hot topic even for offline spike sorting algorithms (Pachitariu et al., 2016; James J Jun et al., 2017; Shan et al., 2017). While no common answer to this problem do exists, the denser the probe, the easier it is to potentially track the neurons over time, and appreciate the distortions of the spike waveforms over time. This is why we designed a simple but modular framework allowing the dictionary of templates to be modified over time. While currently the tracking of the templates is rather simple, one could think about more clever ways to compensate for continuous drifts.

Closed-loop experiments with online spike sorting would open the door to several new protocols. For example, it has been shown (Benda et al., 2007) how adaptive sampling (instead of naïve, brute force sampling of a given parameter space) could drastically optimize the design of some key standard stimulation protocols that allows the researcher to identify or characterize a sensory neuron, at the beginning of an experiment. In the visual system, this may be the receptive field of a neuron, in the auditory system its threshold curves over sound frequencies. A study assuming closed-loop spike sorting (Lewi et al., 2007), shows how to measure receptive fields more efficiently but more importantly suggests an approach to do so in an optimal and most time-saving way.

3.4 Material and methods

3.4.1 Code

The code is written as a Python package, available as an open source software `https://github.com/spyking-circus/spyking-circus-ort`. By default, all blocks behave similarly to what have been described in (Yger et al., 2018), since the code is built on the same template-matching based algorithm. Com-

munications between `blocks` is based on the ZeroMQ library (Hintjens, 2010) allowing a fast and efficient scheme for message passing, either synchronous or asynchronous.

We advise the reader to refer to the previously mentioned paper in order to get the fine details of the algorithm, and only the key differences will be summarized and/or highlighted in the following. We consider N_{elec} channels, acquired at a sampling rate f_{rate}. Every channel k is located at a physical position $\mathbf{p}_k = (x_k, y_k)$ in a 2D space with $k \in \{1, \cdots, N_{\text{elec}}\}$ (extension to 3D probes would be straightforward). The aim of our online spike sorting algorithm is to decompose the signal $\mathbf{s}$ defined over time and over all channels as a linear sum of spatiotemporal kernels or "templates" such that:

$$\mathbf{s}(t) = \sum_{i,j} a_{i,j}\mathbf{T}_j(t - t_i) + \mathbf{e}(t) \tag{3.1}$$

where $\mathbf{s}(t)$ is the signal recorded over all the electrodes at time t, $\mathbf{T}_j$ is the template associated to cell j which represents the waveforms triggered on all the electrodes when this cell elicit an action potential, $\{t_i\}_i$ represents the putative spike times of the cell population, $a_{i,j}$ corresponds to the amplitude factor for the ith spike time and jth cluster (note that this value can be equal to 0), and $\mathbf{e}(t)$ is the background noise. The algorithm can be divided conceptually into two main steps, described below. After a preprocessing stage, we first run a clustering algorithm to extract a dictionary of "templates" from the recording. Second, we use these templates to decompose the signal with a template-matching algorithm. We assume that a spike will only influence the extracellular signal over a time window of size N_t (typically 3 ms for *in vivo* and 5 ms for *in vitro* data), and only electrodes whose distance to the soma is below r (typically 100 µm for *in vivo* and 200 µm for *in vitro* data). For every channel k centered on $\mathbf{p}_k$, we define N^k_{neigh} as the ensemble of nearby channels such that $\forall p \in N^k_{\text{neigh}}$, $\|\mathbf{p} - \mathbf{p}_k\|_2 \leq r$. In the following, we note $\mathbf{s}[t]$ the discrete-time signal obtained after sampling of the continuous-time signal $\mathbf{s}(t)$, and we note $\mathbf{s}_k[t_{\min} : t_{\max}]$ the temporal slice of the signal recorded on channel k between $t_{\min}$ and $t_{\max}$.

Below are the main building blocks of the algorithms (turned into independent computational `blocks`), with algorithmic details for each of them.

3.4.2 Filtering

The channels were first individually high-pass filtered with a Butterworth filter of order three and a cutoff frequency of 300 Hz, to remove any low-frequency components of the signals. We then subtracted, for every chan-

nel k, the median such that $\text{med}_t\ s_k[t] = 0$, where med_t is the median over time.

3.4.3 Whitening

To remove spurious spatial correlations between nearby recordings channels, we performed a spatial whitening on the data. To do so, we collected 20 s of recordings and computed the covariance matrix of the noise $\mathbf{C}_{\text{spatial}}$, and estimated its eigenvalues $\{d_k\}_k$ and associated eigenvectors. From the diagonal matrix $\mathbf{D} = \text{diag}(1/\sqrt{d_1 + \epsilon}, \cdots, 1/\sqrt{d_{N_{\text{elec}}} + \epsilon})$, where $\epsilon = 10^{-18}$ is a regularization factor to ensure stability, and from the square matrix $\mathbf{V}$ whose columns are the eigenvectors, we computed the whitening matrix $\mathbf{W}$ as $\mathbf{VDV^T}$. In the following, each time chunks of data are loaded and then multiplied by $\mathbf{W}$.

3.4.4 Threshold estimation and peak detection

We computed a spike threshold θ_k for every channel k such that $\theta_k = \alpha\ \text{MAD}_t\ s_k[t]$, where MAD_t is the Median Absolute Deviation over time, and α is a free parameter. We always used $\alpha = 6$. For every channel k, we detected some putative spike times $\{t_{i,k}\}_i$ which correspond to all the local minima of $\mathbf{s}_k$ below $-\theta_k$.

3.4.5 Basis estimation (PCA)

To identify the spatiotemporal waveforms embedded in the data, we need to reduce their dimensionality. For each channel k, we collected up to N_{w} spikes at times $\{t_{i,k}\}_i$. In order to compensate for sampling rate artifacts, we upsampled all the collected waveforms by bicubic spline interpolation to 5 times the sampling rate f_{rate}, aligned on their local minima, and then re-sampled at f_{rate}. We performed a Principal Component Analysis (PCA) on these centered and aligned waveforms and kept only the first N_{PCA} principal components. In all the calculations, we used default values of $N_{\text{w}} = 1000$ and $N_{\text{PCA}} = 5$. These principal components were used during the clustering step.

3.4.6 Density-based clustering

In order to parallelize the problem, we used a divide and conquer approach (Marre et al., 2012; Swindale and Spacek, 2014). Each time

a spike was detected at time $t_{i,k}$ on electrode k, we searched for electrode ϕ where the voltage has the lowest value, i.e. such that $\phi = \arg\min_{k'} \mathbf{s}_{k'}[t_{i,k} - N_t/2 : t_{i,k} + N_t/2]$. We replaced $t_{i,k}$ by $t_{i,\phi}$ in the set of collected spike times.

So for every electrode k we collected spikes peaking on this electrode. Each of these spikes was represented by a spatiotemporal waveform of size $N_t \times N^k_{\text{neigh}}$. Note that this pre-grouping does not assume that the spikes were only detected on a single electrode. This clustering performed on each spike ensemble used the information available on all the neighboring electrodes. We projected each waveform on the PCA basis estimated earlier to reduce the dimensionality to $N_{\text{PCA}} \times N^k_{\text{neigh}}$. During this projection, the same upsampling technique described in the previous section was used. For every channel k, we collected N^k_{spikes} spikes, and each of them was a vector of size $N_{\text{PCA}} \times N^k_{\text{neigh}}$. The maximal number of spikes collected is defined by the user as N_{spikes}, and we used a default value of $N_{\text{spikes}} = 1000$.

To reduce dimensionality even further before the clustering stage, for every channel k we performed a PCA on the collected spikes, and kept only the first N_{PCA_2} principal components (in all the paper, $N_{\text{PCA}_2} = 10$). Therefore, we performed a clustering in parallel for every channel, on at max N_{spikes} described in a space of N_{PCA_2}-dimension.

Every electrode k performed a local density-based clustering, as described in (Yger et al., 2018). If the software is use in an hybrid "offline/online" mode (see figure 3.5A), then no major difference are noticeable. As soon as $N_{\text{spikes}} = 2000$ are collected for each channel, then the density-based clusterings are launched, and the found templates are stored in a static dictionary used afterwards during the template matching procedure (see the two following sections). The major difference arise from the extension to data stream, if activated by the user. In such a mode, every cluster is turned into a core macro-cluster, and the dictionary of templates will be modified during the course of the experiment (see Online clustering section).

3.4.7 Template estimation

At the end of the clustering phase, we pooled the clusters obtained from every electrode and we obtained for every cluster j a list of spike times $\{t_{i,j}\}_i$. We computed the template from the raw data as the pointwise median of all the waveforms belonging to the cluster such that $\mathbf{T}_j(t) = \text{med}_i\, \mathbf{s}(t - t_{i,j})$ where med_i is the median taken over the collected spikes. Note that $\mathbf{T}_j$ is only different from zero on the electrodes close to its peak. This information is used internally by the algorithm to save templates as sparse structures. Moreover, during template estimation, we limited the number of spike times per tem-

plate to a maximal value of 200 to avoid memory saturation. To enhance the compression level of the template $\mathbf{T}_j$, we set to 0 all the channels k where $\forall t,\ |T_{j,k}(t)| < \theta_k$, with θ_k the detection threshold of channel k. This allowed us to remove channels without discriminant information, and to increase the sparsity of the templates. Once the template $\mathbf{T}_j$ had been extracted, we computed its minimal and maximal amplitudes $a_{\min,j}$ and $a_{\max,j}$ based on data used during the clustering, i.e. spike times limited to those collected for this step. If $\hat{\mathbf{T}}_j$ is the normalized template, such that $\hat{\mathbf{T}}_j = \mathbf{T}_j / \|\mathbf{T}_j\|_2$, we computed:

$$
\begin{aligned}
a_{i,j} &= (\mathbf{s} \star \hat{\mathbf{T}}_j)(t_{i,j}) = \int \mathbf{s}(t - t_{i,j}) \hat{\mathbf{T}}_j(t) dt \\
a_{\min,j} &= \text{med}_i\ a_{i,j} - 5\ \text{MAD}_i\ a_{i,j} \\
a_{\max,j} &= \text{med}_i\ a_{i,j} + 5\ \text{MAD}_i\ a_{i,j}
\end{aligned}
$$

where $\mathbf{s} \star \hat{\mathbf{T}}_j$ is the cross-correlation function between the signal and the normalized template, and MAD_i is the MAD taken over the collected spikes. Those boundaries are used during the template matching phase (see Template fitter section).

3.4.8 Template updater

While templates are found by the independent density-based clustering, they are sent to the template updater, in charge of maintaining an exhaustive dictionary of all the templates $\{\mathbf{T}_j\}_j$, and remove putative duplicates due to the divide and conquer approach. For example, a neuron in between two electrodes would give rise to two very similar templates, one for each of these electrodes. Thus, for each newly incoming template $\mathbf{T}$, we computed its similarity with the ones already available in the dictionary with $CC_{\max}(\hat{\mathbf{T}}, \hat{\mathbf{T}}_j) = \max_\tau\ (\hat{\mathbf{T}} \star \hat{\mathbf{T}}_j)(\tau)$, which find the lag τ such that the cross-correlation between the templates is maximized. If $CC_{\max}(\hat{\mathbf{T}}, \hat{\mathbf{T}}_j) \geq CC_{\text{similar}}$, we considered this new template to be redundant and it is ignored. In all the following, we used $CC_{\text{similar}} = 0.975$. Note that we are computing the cross-correlations between normalized templates, such that two templates that have the same shape but different amplitudes are merged. As opposed to the offline implementation of SpyKING CIRCUS, we currently can not afford to spot the putative mixtures of templates online, since this is computationally prohibitive. However, in the online context we could easily use a method similar to the one described in (Lee et al., 2020): clusters with a high temporal variability are likely to be mixtures. This relies on the hypothesis that if two cells are firing together, the probability of

having the two spikes exactly aligned is lower, and thus the temporal variability of the snippets should on average be higher for such mixtures than for isolated waveforms.

3.4.9 Template fitter

Combining the dictionary of templates $\{\mathbf{T}_j\}_j$ and all the putative spike times $\{t_i\}_i$, we now need to reconstruct the signal $\mathbf{s}$ by finding the amplitudes coefficients $a_{i,j}$ described in equation 3.1. Note that most $a_{i,j}$ in this equation are equal to 0. For the other ones, most $a_{i,j}$ values are around 1, because the corresponding spike at time t_i usually appears on electrodes with a shape and an amplitude similar to template $\mathbf{T}_j$. In this template matching step, all the other parameters have been determined by template extraction and spike detection, so the purpose is only to find the values of these amplitudes. To do so, we used an iterative greedy approach to estimate the $a_{i,j}$ for subgroups of successive putative spikes, which bears some similarity to the matching pursuit algorithm (Mallat and Z. Zhang, 1993). The main idea is to find the pair (i, j) which maximizes $(\mathbf{s} \star \mathbf{T}_j)(t_i)$, then to set the amplitude $a_{i,j}$ equal to $(\mathbf{s} \star \hat{\mathbf{T}}_j)(t_i)$, to check if the amplitude value is between $a_{\min,j}$ and $a_{\max,j}$, and finally to subtract the scaled template $a_{i,j}\,\mathbf{T}_j$ shifted on t_i from the signal $\mathbf{s}$. See (Yger et al., 2018) for more details.

3.4.10 Online clustering

In order to deal with online modification of the dictionary of templates, we adapted our clustering algorithm to the concept of micro-clusters, as described in (Amini and Wah, 2011). More precisely, a micro cluster is a temporal extension of cluster feature **CF** explained in (T. Zhang et al., 1996), that is a summarization triple maintained about a cluster. We have $\mathbf{CF} = (N, \mathbf{LS}, SS)$ where N is the number of data points in the cluster, $\mathbf{LS} = \sum_i^N \mathbf{x}_i$ is the linear sum of these data points, and $SS = \sum_i^N x_i^2$ is their squared sum. The methods are similar to what is done in the DenStream algorithm (Cao et al., 2006), where the micro cluster extends the **CF** vector into the temporal domain, and each data point is weighted by a memory function $f(t) = 2^{-\lambda t}$ where $\lambda > 0$. To be more explicit, for data points $\{x\}_i$ collected at timestamps $\{T\}_i$, a micro cluster is defined by the tuple (w, c, r) where:

$$w = \sum_i^N f(t - T_i) \tag{3.2}$$

$$c = \frac{\sum_i^N f(t - T_i) x_i}{w} \tag{3.3}$$

$$r = \frac{\sum_i^N f(t - T_i)\|x_i - c\|_2}{w} \tag{3.4}$$

w is the weighted sum of all the fading factors, c is the centroid of the cluster, taken into account the fading memory, and r is the radius of the cluster, i.e. its spatial extent. Basically, micro clusters are divided into two categories, based on a density criteria. If $w < \beta\mu$, the cluster is defined as an outlier micro cluster, and if $w > \beta\mu$ as a core micro cluster. The terminology is inspired by the DBSCAN clustering algorithm (Ester et al., 1996), where a cluster is said to be dense when the weight w is higher than a value μ and the radius r is below a fixed radius ϵ. In all our simulations, we fixed $\mu = 2$ and ϵ as the averaged radius of the first clusters obtained, on a given electrode. The decay parameter for the fading function is fixed to $\lambda = 10$s. The β parameter is set to 0.5.

In order to discover the clusters in an evolving data stream (the incoming spikes), we maintain a group of core micro clusters and outliers micro-clusters in an online way. All the outliers micro clusters are maintained in a separate memory space, say an outlier buffer. When a new point x arrives (a spike), the procedure of merging is described below (see Algorithm 1 for details in (Cao et al., 2006)).

1 At first, we try to merge x into its nearest core micro cluster c_{core}. If r_{core}, the new radius of c_{core}, is below or equal to ϵ, merge x into c_{core}.

2 Else, we try to merge x into its nearest outlier micro cluster c_{outlier}. If r_{outlier}, the new radius of c_{outlier}, is below or equal to ϵ, merge x into c_{outlier}. And then, we check w the new weight of c_{outlier}. If w is above $\beta\mu$, it means that c_{outlier} has grown into a core micro cluster Therefore, we turn $c_{outlier}$ into a new core micro cluster

3 Otherwise we create a new outlier micro-cluster c_{outlier} by x. This is because x does not naturally fit into any existing micro cluster. It means that x may be an outlier or the seed of a new micro-cluster.

Pruning mechanisms are implemented such that outliers micro clusters that do not get updated regularly are deleted, to keep the memory consumption low (Cao et al., 2006). When enough new incoming spikes have been detected (we usually use a fixed number $N_{\text{spikes}} = 2000$, we performed a density-based clustering similar to the one done in (Yger et al., 2018) on the core micro clusters, that have been maintained dynamically during the arrivals of the spikes. This clustering (at time t_1) results in a list of clusters, that can be compared to the ones obtained in the previous clustering (obtained at time

Block	Inputs	Outputs
`writer`	raw data	
`filtering`	raw data	filtered data
`whitening`	filtered data	whitened data
`threshold`	whitened data	thresholds
`detector`	whitened data, thresholds	peaks
`clustering`	whitened data, peaks	templates
`template updater`	templates	
`template fitter`	whitened data, peaks, templates	spikes
`spikes writer`	spikes	

Table 3.1: List of the main computational `blocks` available to perform online spike sorting, with their default inputs and outputs.

t_0). And here, we used a simple rule to ensure that clusters drifting over time are "tracked". For every new cluster obtained at time t_1, we checked if a centroid seen at t_0 could be seen in its vicinity, given a radius ϵ. If this is the case, then we update the template, and otherwise, a new template is added in the dictionary.

3.4.11 Synthetic dataset

Generation of the templates

In order to test the performance of the software, we developed a modular framework, somehow similar to the one described in (Buccino et al., 2019). This framework was meant to benchmark all the possible situations encountered with online spike sorting. In an abstract manner, users can create `cells` objects with time varying properties, such as firing rates, amplitudes and positions. More precisely, the waveform $v_k(t)$ triggered at a given channel k at time t_i by one cell is defined with a simple formula:

$$v_k(t) = \frac{a}{1 + (d/\delta)^2} \sin(4\pi(t - t_i)) \Big((t - t_i) \mathrm{e}^{-(t-t_i)/\tau} \Big)^{10} \tag{3.5}$$

where a, δ and τ are fixed parameters modeling the attenuation of the waveforms as function of time and distance with the nearby channels, and d is the euclidean distance between the soma of the cell (at position $(x(t), y(t))$) and the position of the recording site for channel k. By default, we used $a = -80$ µV, $\delta = 45$ µm and $\tau = 1.5$ ms. In order to enforce the uniqueness of every spatiotemporal template, a sparsification factor ϵ was introduced to control

the proportion of the channels, selected randomly, that will have non-zero waveforms. We used $\epsilon = 0.5$.

Generation of the spike trains

Spike trains are generated as inhomogeneous Poisson spike Trains with a time varying firing rate $r(t)$. While in the current paper, this benchmark pipeline had all the ingredients to assess the quality of online sorting in the context of drifts/change of dynamics during the course of an experiment, we used only fixed positions and firing rates in the current paper.

Generation of the noise

Finally, once templates and spike trains are created, we superimpose them onto a Gaussian noise $\mathcal{N}(0, \sigma^2)$ with $\sigma^2 = 4$ µV.

3.4.12 Error rate

The error rate is computed as the mean between the False Negative Rate (FNR) and Positive Predicted value (PPV). FNR is defined as the proportion of all positives, true positives (TP) and false negatives (FN), that were flagged as false negatives:

$$FNR = \frac{FN}{FN + TP} \tag{3.6}$$

Similarly, the PPV is the proportion of positive results, false positives (FP) and true positives (TP), that are indeed true positives (TP):

$$TP = \frac{TP}{FP + TP} \tag{3.7}$$

Combining these two quantities, we defined the error rate ϵ as:

$$\epsilon = \frac{FNR + (1 - PPV)}{2} \tag{3.8}$$

3.4.13 Ground truth recordings

We used the already published dataset (Spampinato et al., 2018) for the ground truth recordings. More precisely, electrophysiological recordings were obtained from *ex-vivo* isolated retinas of rd1 mice (4/5 weeks old). The retinal tissue was placed in AMES medium (Sigma-Aldrich, St Louis, MO; A1420) bubbled with 95 % O2 and 5 % CO2 at room temperature, on a MEA (10 µm electrodes spaced by 30 µm; Multichannel Systemps, Reutlingen,

Germany) with ganglion cells layer facing the electrodes. Borosilicate glass (BF100-50, Sutter instruments) electrodes were filled with AMES with a final impedance of 6-9 MΩ. Cells were imaged with a customized inverted DIC microscope (Olympus BX 71) mounted with a high sensitivity CCD Camera (Hamamatsu ORCA -03G) and recorded with an Axon Multiclamp 700B patch clamp amplifier set in current zero mode.

Chapter 4

A perturbative approach to the retina

We have seen that online, automated and scalable spike sorting of high-throughput electrophysiological data is accessible. This open the possibility to perform closed-loop experiments to dynamically probe the spiking activity of neurons. For example, this could significantly reduce the number of trials needed to characterize neural responses by choosing the most informative stimulus at each experimental step (Lewi et al., 2007).

Here, I present a specific application on the retina where online spike sorting might be useful. Classically, retinal ganglion cells are supposed to extract specific features from the visual image even if this processing is known to be dependent on visual context. I study the stability of feature extraction in ganglion cells in different natural contexts sharing the same mean luminance and contrast. In particular, I show that some cells can respond selectively to increases of light in some contexts and to decreases in others, and describe a model that can recapitulate this effect.

This chapter corresponds to an article in preparation in collaboration with Ulisse Ferrari, Alexander S Ecker, Yannick Andéol, Olivier Marre and Thierry Mora.

4.1 Introduction

Primates are able to recognize objects effortlessly (Logothetis and Sheinberg, 1996; DiCarlo et al., 2012). However multiple sources of variation in their visual appearances make this task computationally difficult. For example, a change in the orientation relative to the observer results in extremely different input images for the visual system. In other words, object recognition solves an important *invariance problem*.

To achieve object recognition, it has been proposed that each area of the visual system extracts features from the visual scene, and that these features are more and more abstract, allowing object recognition in higher order areas. This feature extraction starts in the retina. Retinal ganglion cells, the output of the retina, can be classified in different cell types (Baden, Berens, et al., 2016). Each of them is supposed to extract one low-level visual feature (Silveira and Roska, 2011). A classical example is the first-order classification of ganglion cells between ON and OFF types. This distinction relies on their response to step-like changes in luminance. ON cells are sensitive to light increments whereas OFF cells are sensitive to light decrements. This property is referred to as *polarity* and is considered as one of the basic features extracted by retinal ganglion cells.

However, this feature extraction seems to depend on the visual context. For example, Tikidji-Hamburyan et al. (2015) showed that the polarity is not fixed but changes qualitatively for different background luminance. They reported that over 80% of the cells had opposite polarities at different light levels. A following study from Pearson and Kerschensteiner (2015) even identified three functional types that switch polarity in a luminance-dependent manner. Transient variations of the polarity have also been reported by Geffen et al. (2007). In this case, a grating of dark and light bars was used as a peripheral image. Rapid contrast inversions of this grating were sufficient to trigger the same effect. Over 16% of the cells presented this kind of polarity reversal.

However, in these studies, retinal processing was mostly probed with artificial stimuli. It is not clear if the retinal code is also context-dependent during natural stimuli, provided that contrast and luminance are stable. Recently, Maheswaranathan, McIntosh, Kastner, et al. (2018) reported predictions from deep learning models suggesting that retinal processing, and in particular the polarity of ganglion cells, can quickly change during natural scene stimulation. However, this was not tested experimentally.

Here we introduce a new experimental approach to investigate context-dependent processing during natural stimulation, by measuring context-dependent receptive field. We asked: given an average luminance and con-

trast, are ganglion cells stimulated with natural images always selective to increments or decrements of light no matter what the natural images are?

To answer this question, we adopted a perturbative approach. We stimulated ganglion cells with natural images, and then added small checkerboard-like perturbations on top of them. These perturbations evoked small changes on the responses of retinal ganglion cells. For each cell and each image, we estimated the perturbation that evoked on average the largest increase of the response. We referred to this quantity as the *local spike triggered average* (LSTA). We found that the LSTAs were highly dependent on the images. For some cells, we even observed changes in polarity for different images. This means that these cells could be sensitive to light increments for one natural image and sensitive to light decrements for another one. We found a nonlinear model that can account for these findings. This high dependence on context suggest that luminance is probably not the feature which is reliably extracted by these cells, but we found that this context dependence was compatible with a robust extraction of contrast.

4.2 Results

4.2.1 A new experimental procedure to estimate receptive field locally

We created a visual stimulus to probe the receptive field of one cell in a natural context. This stimulus is composed of a sequence of grayscale images and has been presented to isolated salamander retinas using a digital projector. Each image has been flashed for a given period of time Δt (fixed to 300 ms since (Thorpe et al., 1996) reported that the human visual system can analyse complex stimuli in less than 150 ms). We also added the same period of time between each succession of flashes during which a uniformly gray image was presented. This last precaution is necessary to make each neural response to one flashed image independent from the image flashed just before. Figure 4.1A (left) illustrates this stimulus design.

To generate perturbed natural images, we picked a reference image in the Van Hateren dataset (Van Hateren and Schaaf, 1998), which provides a strictly linear relationship between scene luminance and pixel value. We used a normalization step to fix the mean luminance and the root mean square (RMS) contrast of this reference image. We then added a dim checkerboard pattern composed of randomly black and white checks, with no spatial structure, and a small contrast relative to the natural image (figure 4.1A, right). By adding this perturbation, the parts of the image where a “black” check

was added became slightly darker. The parts where a "white" check was added became slightly brighter. We generated many perturbed images from several reference images. All the images were randomly shuffled.

We recorded the activity from ganglion cells using microelectrode arrays (MEAs) and collected the responses of each cell to each flashed image. For each cell and each reference image, we estimated the spike-triggered average of the perturbation patterns, defined as the average perturbation pattern that elicited a spike. This is the sum of the perturbation pattern weighted by the number of elicited spikes, divided by the total number of spikes. We thus obtained a "local'' spike-triggered average (named hereafter LSTA) for each cell and each reference image (figure 4.1B). In the following we referred to these reference images as the "natural contexts''. Examples of two LSTAs and their corresponding perturbed image are shown in figure 4.1C.

How does the analysis described previously relates to the white noise analysis introduced in (Chichilnisky, 2001)? The two analyses are quite similar however two differences are worth highlighting. First, the choice of the reference image is different. In (Chichilnisky, 2001) this is the origin in the stimulus space and a spatially uniform gray image of mean intensity. We relaxed this constraint and considered that natural images are eligible as reference images. As a control, in the following we also used a spatially uniform gray image as a reference image. Second, the perturbation amplitude is different. In (Chichilnisky, 2001) this corresponds to the contrast specified to generate a white noise sequence. This contrast is rather high to maximize the neural response. In our case, the perturbation amplitude is the lowest that can still evoke some changes in the response. If the amplitude is too low, we will not see any change in the response compared to the response to the reference image. As a result, the LSTA will be flat. If the amplitude is too high, the contrast of the checkerboard will be much higher than the contrast of the image. As a result, the response would be dominated by the checkerboard pattern, and the LSTA would not depend on the reference image. We chose an amplitude that was between these two extremes.

4.2.2 Retinal ganglion cells can be sensitive to opposite polarities in different natural contexts

For multiple ganglion cells, the LSTA changed polarity depending on the reference image. In particular, some LSTAs were either ON or OFF depending on the natural context. For example, the cell illustrated in figure 4.2 can been clearly identified has an OFF cell based on its estimated spatial and temporal STAs (checkerboard stimulus). In other words, this cell is thought to increase

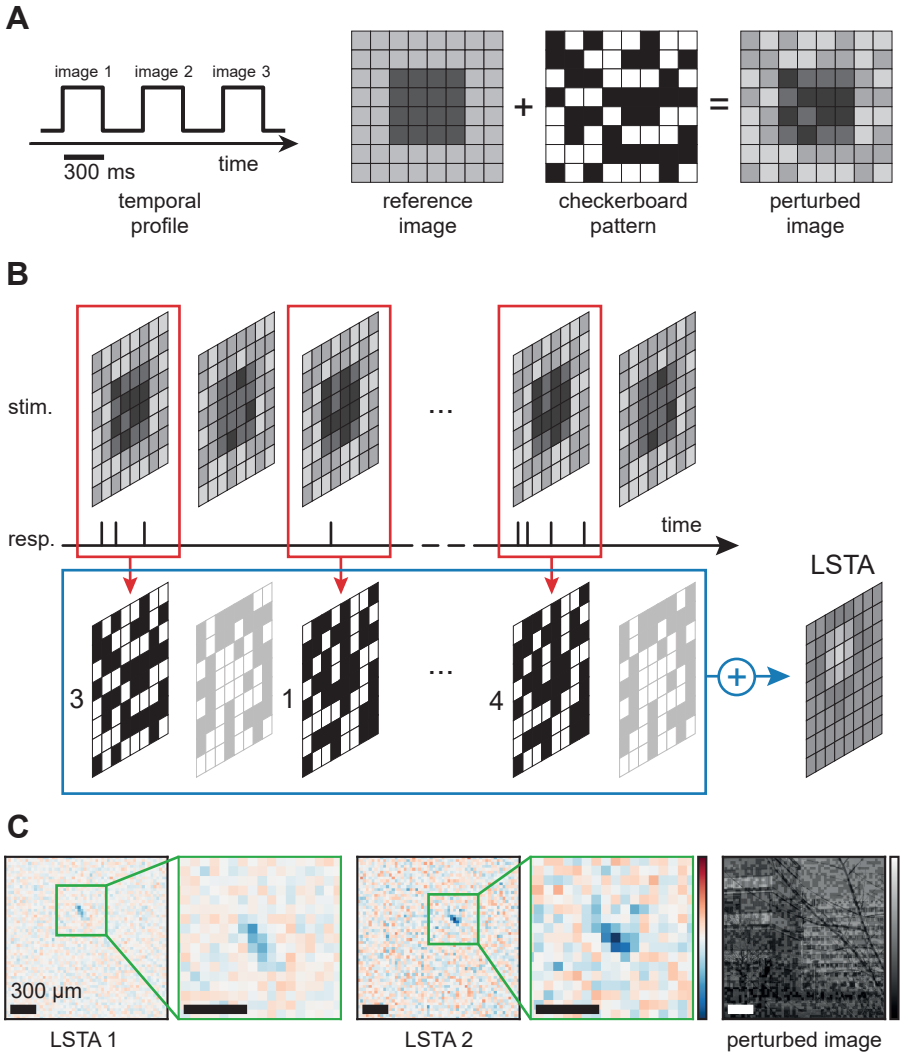

Figure 4.1: Stimulation protocol to measure LSTAs. **A.** Stimulus design. We flashed a randomized sequence of perturbed images. All flashes lasted 300 ms and were separated by 300 ms of gray equal to the mean luminance. Perturbed images were generated by adding randomized checkerboard patterns with lower contrast to reference images. **B.** LSTA calculation. The sequence of perturbed images was presented, and spikes from the neuron were recorded. The spikes in some time window corresponding to flashes (red) were counted to weight the corresponding checkerboard patterns. The weighted average (blue) gave the LSTA which indicated the small perturbation that can change the most neuronal responses. **C.** Two example LSTAs, measured for two example cells and the same perturbed image, show clear OFF centers located at different positions.

spiking activity in response to light decrements and not to light increments. However we found that LSTAs were dramatically different across contexts for this cell. The LSTA associated to the control context (i.e. the "gray" image) was similar to the global STA in terms of polarity — both were OFF. However, for two natural contexts (i.e. "building" image and "umbrella" image) the LSTAs had an ON polarity. In these contexts, the same cell will increase its spiking activity in response to an increase of luminance inside its receptive field center. The LSTA of the last context (i.e. "branches" image) was found to have OFF polarity. In summary, the polarity of the LSTA of this cell in different natural contexts differed qualitatively.

The same observations can be made on figure 4.3 but for a different ganglion. This cell is also identified as an OFF cell (figures 4.3B-C) but, for three of the fourth contexts, LSTAs were measured with ON polarity (figure 4.3E). These contexts can also be distinguished by the low level of neural activity evoked by all the perturbed images (figure 4.3F). Their small number of elicited spikes made the estimation of their LSTAs more noisy but clear spatial structures could still be observed in the receptive field center. We observed that these spatial structures were displaced compared to the spatial STA (green ellipse) and were different from one context to the other. This suggests that the position and shape of the LSTA of this cell vary across natural contexts.

During preliminary experiments, a higher mean luminance have been tested to measure the LSTAs. Figure 4.4 shows that some inversions of the polarity could also be observed with two LSTAs having ON polarity (figure 4.4B). Interestingly, these two contexts are associated with similar raster plots (figure 4.4C) which shows a peak of the firing rate around 100 ms after the onset of the stimulus. The other raster plots show a peak of activity which occurs earlier, 50 ms after the onset. In this case, the latency of the response correlates with the polarity of the LSTA. This suggests that two different pathways are at play since this difference in latency could be explain by their difference in length (number of synapses to be crossed). This observation has not been explored in more details, and a lower mean luminance is assumed in the rest of this chapter.

4.2.3 Linear and nonlinear models

To better explain this change of polarity, we fitted linear and nonlinear regression models to predict the activity from the stimulus, as well as the LSTAs. The inputs to the regression models were unperturbed natural images which were briefly flashed on the retina (figure 4.5A; figure 4.5B, first row). The linear regression model was a conventional linear-nonlinear model

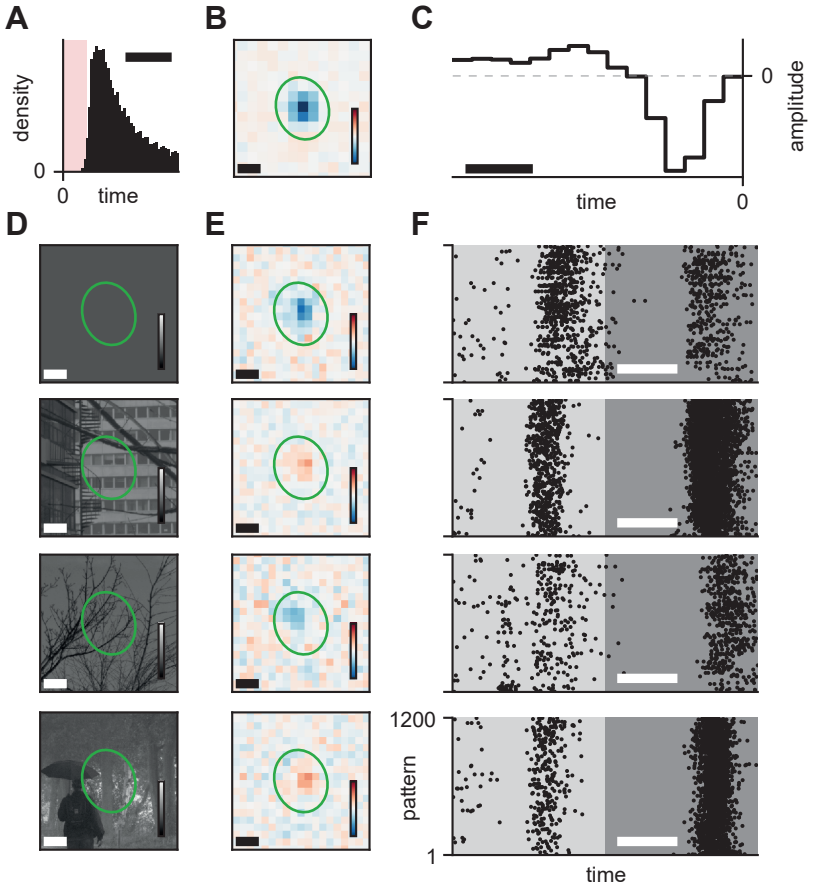

Figure 4.2: Variations of LSTAs over natural contexts (example cell 1). **A.** Interspike-interval histogram indicating a rather low variability of neural responses (5 ms refractory period violation: 0.4 %, scale bar: 10 ms, bin size: 0.5 ms). **B.** Spatial spike-triggered average as a linear estimate of the spatial profile of the receptive field (scale bar: 100 µm). The ellipse (green) represents the position and shape of the spatial receptive field (3 standard deviation contour of the 2D Gaussian fit). **C.** Temporal spike triggered average as a linear estimate of the temporal profile of the receptive field (scale bar: 100 ms). **D.** Reference images tested as contexts. (scale bar: 100 µm). **E.** LSTAs measured experimentally for each context. (scale bar: 100 µm). **F.** Rasters for each context (scale bar: 100 ms). Each line corresponds to a randomized checkerboard pattern and each point corresponds to a spike. The light gray area corresponds to the stimulus time (i.e. when the perturbed images were flashed), and the dark gray region to the inter-stimulus time (i.e. uniform gray displayed).

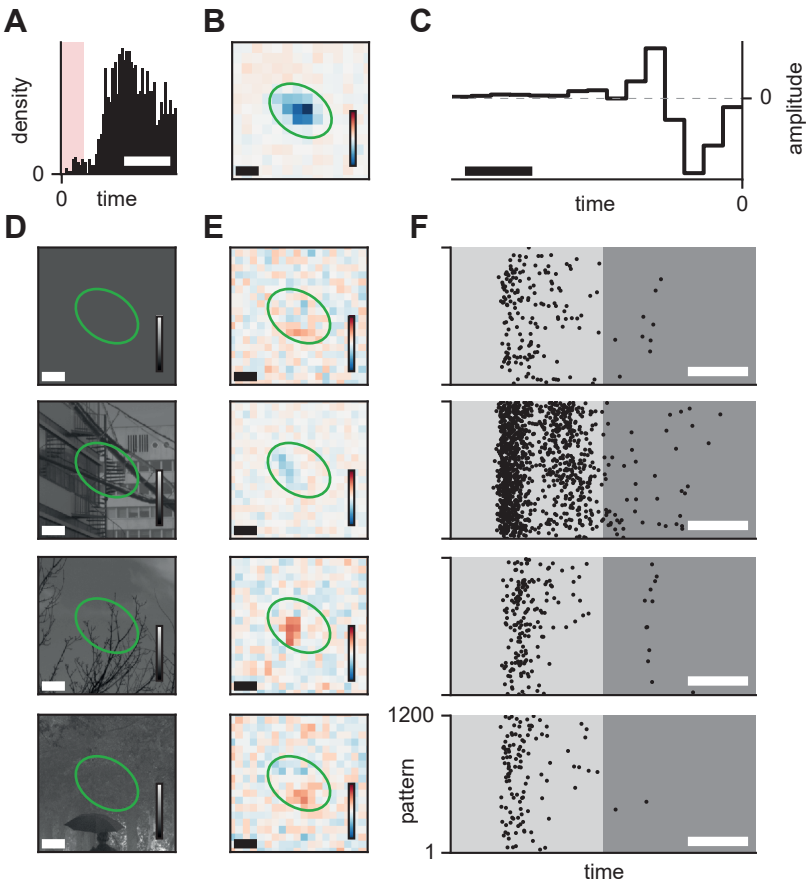

Figure 4.3: Variations of LSTAs over natural contexts (example cell 2). Same legend as figure 4.2 for a different ganglion cell.

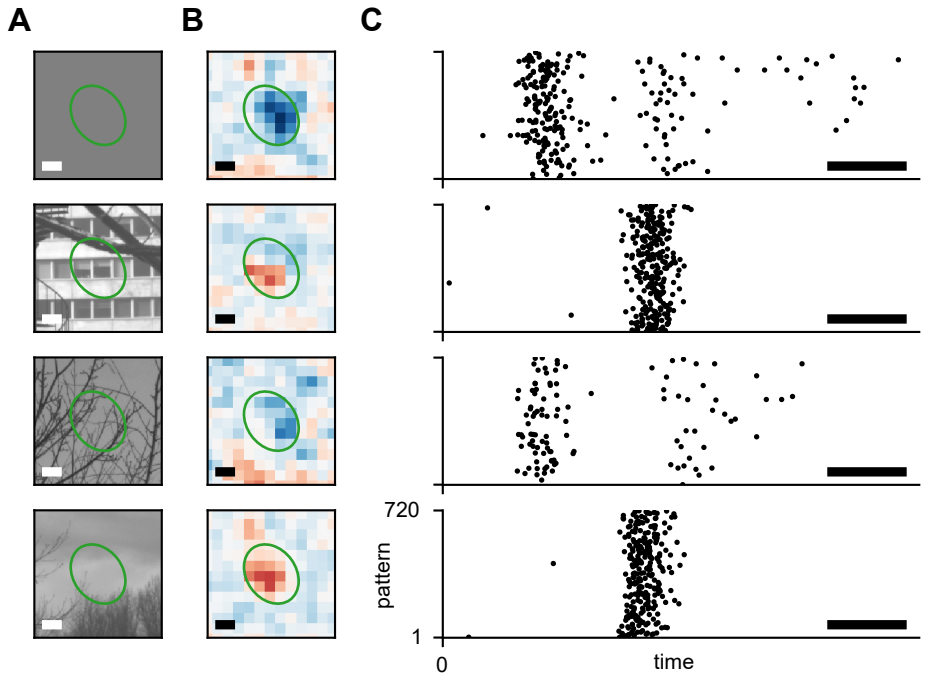

Figure 4.4: Same representation of LSTAs, corresponding reference images, and rasters as in figures 4.2 and 4.3 except that rasters only show the stimulus time (scale bars: 100 µm, 50 ms). This ganglion cell was recorded in a preliminary experiment for which the mean luminance was significantly higher. Note the changes in latencies over different natural contexts.

(LN; Chichilnisky, 2001; Paninski, 2004). It filters the visual stimulus linearly, and then use a nonlinear function to transform the result of this filtering into an output neural response (figure 4.5B, second row).

Additionally, we used two nonlinear encoding models implemented using convolutional neural networks (CNN; LeCun et al., 1998). In these networks, a higher-level representation (i.e. feature maps) is extracted from the visual stimulus before summing some of these features to predict neural responses. In both cases, our networks were composed of two layers: a feature extraction layer followed by a readout layer. Each layer was composed of a linear filter followed by a nonlinearity (i.e. ReLU or softplus). For the first nonlinear model (figure 4.5B, third row), the feature extraction stage was a convolutional layer with two 7×7 2D convolutional kernels. One kernel was initialized with a positive 2D Gaussian profile while the other with a negative one. Both kernels were kept constant during the training phase. For the second layer, we used a fully connected layer with a single output unit. Multiple networks were trained independently for each ganglion cell. In the following, we referred to this model as the linear-nonlinear linear-nonlinear model (LN-LN). A similar model has been used by (Deny et al., 2017).

For the other nonlinear model (figure 4.5B, last row), the first layer was a convolutional layer with four 21×21 2D convolutional kernels. Unlike in the previous model, the filters were learned from the data. The readout stage consisted of multiple fully-connected layers placed in parallel, one for each ganglion cell. The weights of each of these layers were factorized in a 2D spatial mask and a vector of feature weights (with one weight for each feature extracted by the first layer) to decrease the number of model parameters (see Material and Methods). This factorization technique has already been successfully applied by (Klindt et al., 2017; Cadena et al., 2019). Note that the filters of the first layer were common to all ganglion cells. A single network was trained for all the ganglion cells. We referred to this model as the convolutional neural network model (CNN) in the following.

For the LN-LN model, all hidden units and the output unit had parametric softplus activation (Dugas et al., 2001) and were optimized during the learning procedure. For the CNN model, we used ReLU activation for the hidden units. Batch normalization was applied to the output of the first layer of the CNN model to make it more stable. For all models, a Poisson error was used as the training loss function (see Materials and methods).

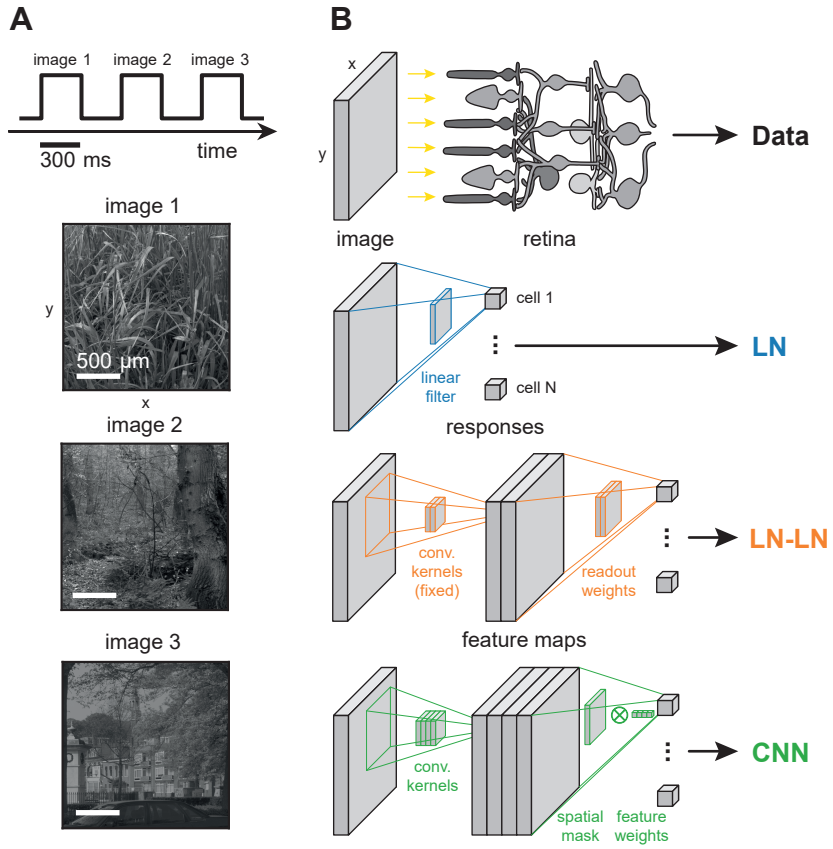

Figure 4.5: Model training. **A.** We used a specific stimulus to train models. We flashed a randomized sequence of natural images (unperturbed). All flashes lasted 300 ms and were separated by 300 ms of gray equal to the mean luminance. We used normalized grayscale natural images taken from the van Hateren's Natural Image Dataset (Van Hateren and Schaaf, 1998) downsampled by a factor of 8 to keep the number of parameters tractable (scale bar: 500 µm). **B.** Schematic of the different architectures used to predict the responses of multiple retinal ganglion cells to a flashed image (first row). The LN model (second row) is formed by a linear filter and nonlinear function (not shown). The LN-LN model (third row) is formed by a convolutional layer (fixed kernels) followed by a dense layer (readout weights). The CNN model (last row) is formed by a convolutional layer (inferred kernel) and a dense layer (readout weights factorized in spatial masks and feature weights). We assumed Poisson noise distribution to determine the cost function to optimize for training and used additional regularization constraints (see Material and methods).

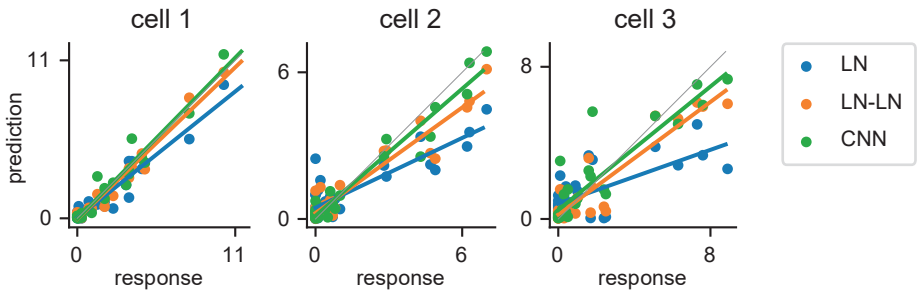

Figure 4.6: Comparison of the responses predicted by the models with the responses observed experimentally for three different cells (responses in spike counts). Each point corresponds to the mean spike count in response to 20 repetitions of the same image. The reported predictions are the mean spike counts output by the models. Each point corresponds to one of the 30 images of the testing set, lines correspond to linear regressions, and colors to models.

4.2.4 Prediction of neural responses using linear and nonlinear models

Examples of observed versus predicted neural responses from the different models are shown in figure 4.6 for three ganglion cells. The predictions of the LN model (blue) were far from the actual responses for some specific cells. This confirms that the LN model can't capture accurately the variations in the neural responses of ganglion cells to flashed natural images. The responses predicted by the LN-LN model (orange) were more similar to the observed responses than the predictions of the LN model. The predictions of the CNN model (green) were even more similar than those of the LN-LN model. This suggests that these nonlinear models are better suited for accurate predictions.

To quantify the absolute performance across all cells, we calculated the prediction accuracy (see Material and methods) between the predicted and observed neural response for each model. Note that even if the model were perfect, the accuracy could still be limited if the observed response is noisy. To correctly estimate the performance, we thus need to compare the accuracy with the best performance reachable given the noise in the observed responses. To quantify the latter we estimated the reliability of these responses (see Material and methods). The reliability corresponds to the maximum proportion of explainable variance. For the LN model, the prediction accuracy is lower than the reliability for a large fraction of the cells (figure 4.7A, first column). This is because these cells encoded the flashed natural images

in a nonlinear manner, not captured by the LN model. We noticed that this was particularly true for cells which have shown clear polarity reversals (red). Figure 4.7A also shows the same comparison for the LN-LN model (second column) and the CNN model (last column). These scatter plots show points closer to the diagonal. This means that the prediction accuracy matches the reliability for most of the cells. In particular, we observed that the CNN model was relevant to predict the neural activity of cells with clear polarity reversals (red).

To better quantify these relative improvements, we used R-squared values instead of simple correlations (see Material and methods). Figure 4.7B shows the R-squared values of the different models on a cell by cell basis along with the limits imposed by the noise, and confirms that the nonlinear models outperformed the LN model. This confirmation is even clearer in figure 4.7C where the noise-corrected R-squared values of the CNN model are compared with those of the LN model (first column). The points deviate significantly from the diagonal. However, note that the prediction performance for the CNN and LN-LN models were comparable (second column). Figure 4.7D summarises performance evaluations at the population level by taking the mean over ganglion cells.

Nonlinear models are thus necessary to make accurate predictions on the stimulus ensemble composed of unperturbed natural images.

4.2.5 Prediction of LSTAs using linear and nonlinear models

The previous section demonstrates that the nonlinear models predicted retinal responses to flashed natural images well. However, it is not clear whether these models could also reproduce the LSTAs obtained for different natural images, and in particular the changes in polarity described previously.

LSTAs describe what would be the linear functions that would best approximate the stimulus-response function locally (Ferrari et al., 2017). As such they are an experimental estimate of the local gradient of this function. To predict the LSTAs with a model, we calculated the gradient of the model output with respect to its input for each ganglion cell and each natural image (see Material and methods). A drawback of this approach is that the gradient is not supposed to be exactly equal to the LSTA, but only proportional to it. In future work we need to estimate the multiplicative constant to make a fully quantitative comparison.

Figure 4.8 shows the comparison between the LSTAs observed experimentally and the ones inferred by each model for a particular cell. The spatial

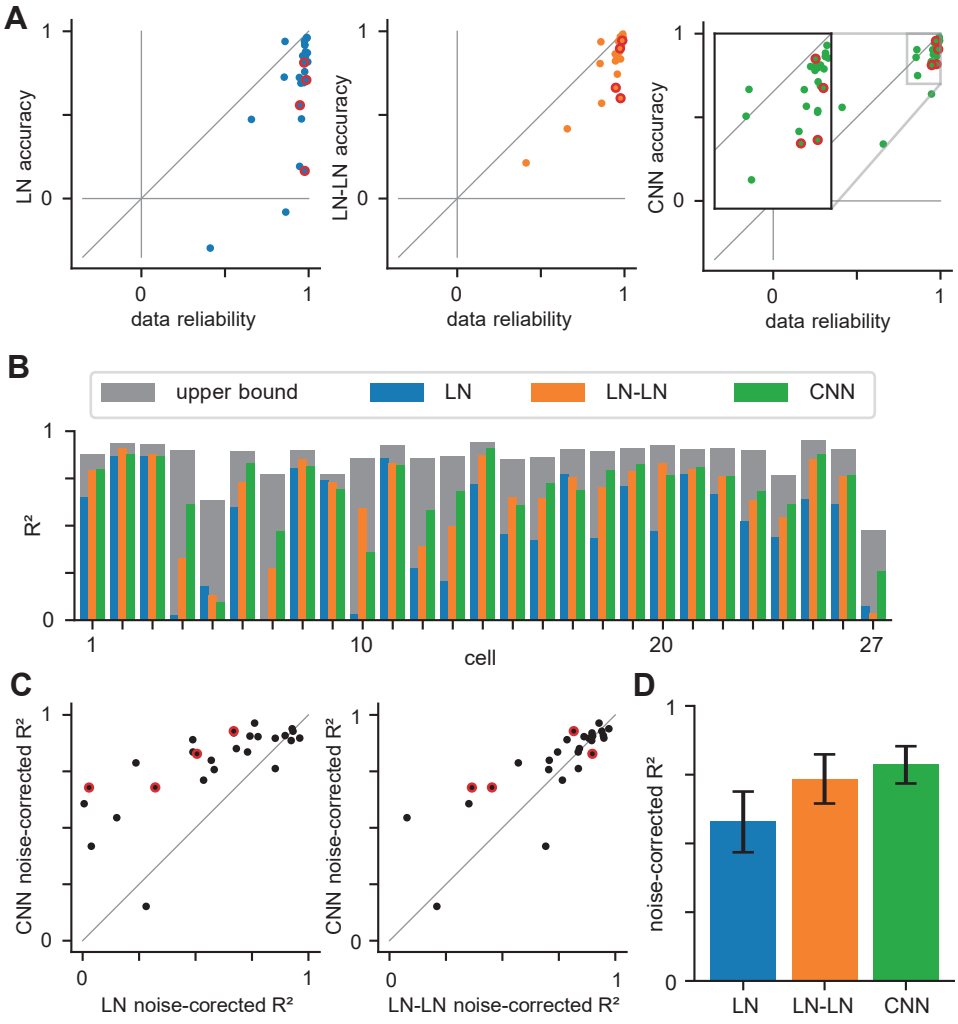

Figure 4.7: Model evaluation. **A.** Comparison between model accuracy and data reliability over the testing set (see Material and methods) for the LN model (first column), the LN-LN model (second column) and the CNN model (last column). Each point corresponds to a ganglion cell. **B.** Comparison of R-squared values between models on a cell by cell basis. Colors correspond to different models and gray represent the maximum R-squared value attainable by a model because of the noisy responses. **C.** Comparison of noise-corrected R-squared values between pair of models: CNN vs LN (first column) and CNN vs LN-LN (second column). Each point correspond to a ganglion cell. **D.** Mean noise-corrected R-squared value over cells for the models (error bars: 95 % confidence interval). **A,C.** Highlighted points (circled in red) correspond to ganglion cells which clearly presented LSTAs of different polarity.

component of the classical spike triggered average estimated with white noise stimulation is shown on the top left corner, and characterizes this cell as an OFF cell. We fitted an ellipse (green) to determine the spatial location of the receptive field of this cell. This ellipse has been reported in all the subsequent plots as a reference. The first column of figure 4.8 also shows the different contexts around the receptive field of the cell. The first image corresponds to the control context (uniform gray) while the three other images correspond to the natural contexts. The second column of figure 4.8 represents the corresponding LSTAs found experimentally. The LSTA for the control context is similar to spatial component of the STA, both in terms of spatial location and polarity. However the LSTAs shown in the second and fourth rows clearly have ON polarity while the LSTA of the third row has OFF polarity. This means that the LSTA polarity of this cell was not the same across tested contexts.

The third column shows the predictions of the LSTAs by the LN model across the different contexts. The most remarkable point is that the prediction was always the same regardless the context. This is due to a fundamental limitation of the LN model: the shape of the gradient is always the same, it can only vary be a multiplicative factor. Thus, this model could not reproduce the observed changes in LSTAs.

The fourth column of figure 4.8 displays the same predictions for the LN-LN model. First, for the control context (first row), we observed that the predicted LSTA was more similar to the experimental one than with the LN model. Second, for the natural contexts (remaining rows), we found a good agreement between the predictions and the corresponding LSTAs. In particular, the polarity was well predicted. However, while the shape was quite similar inside the center of the receptive field (green ellipse), we still noticed that additional structures appear in the surround for two of the contexts (second and fourth rows).

Finally, the last column of figure 4.8 presents the predictions of the CNN model. For each context, predicted LSTAs were very similar to the experimental ones. In contrast with the LN-LN model, no spatial structure was found in the surround of the receptive field. Overall, figure 4.8 shows that the most accurate predictions of the LSTAs were given by the CNN model. This result confirms the superiority of this model to predict both the LSTAs and the neural responses to flashed natural images.

Figure 4.9 shows similar results for another example cell. In this example, the polarity of the LSTA observed for the control context could not be identified clearly (second column, first row) even if the cell was classified as an OFF cell based on its STA (first column, first row). Nevertheless, the prediction of the CNN model (last column, first row) reproduce accurately this

observation by leaving the preferred polarity undetermined. Additionally, the predicted spatial layouts of the LSTAs were more accurate for the CNN model (last column). However, for one context (second row), the predicted LSTA had a particularly high amplitude. This predicts that this cell should have been more sensitive to this particular context. An aspect which was missing from the visualisation of actual LSTAs (second column).

4.2.6 Description of the LSTAs learned by the CNN model

We showed that the CNN model can accurately predict the LSTAs of retinal ganglion cells in different contexts. However the accuracy was assessed on a relatively small number of contexts tested experimentally. This hard constraint was imposed by the limited duration for which we were able to record from a healthy retinal tissue. However, it is important to note that the CNN model wasn't trained on these particular contexts. In other words, this model has the ability to generalize. This validation encouraged us to look at the prediction of the CNN model for additional reference images.

To investigate more largely the variations of LSTAs across contexts, we visually inspected the LSTAs learned by the models over an increased number of natural images. Figure 4.10 shows the predictions given by the CNN model. We observed large variations in the LSTAs depending on the reference image used for the prediction. Furthermore, we clearly identified multiple images for which the LSTAs of the CNN model has ON polarity. Figure 4.11 shows the same predictions for another example cell, with a similar result. This suggests that polarity reversals are more common than what the experimental results suggest.

4.2.7 Interpretation of the components learned by the CNN model

The previous analyses suggest that variations of the LSTAs of ganglion cells in response to flashed natural images is rather the rule than the exception. To understand how these polarity inversions are generated by the CNN model, we inspected its internal components.

To determine the number of pathways necessary to make accurate predictions, we trained multiple CNN model with different number of features. Figure 4.12A shows the noise corrected R-squared for 1 to 4 convolutional kernels. We noticed that the performance doesn't improve beyond 2 kernels hence the optimal number of pathways. The internal components described

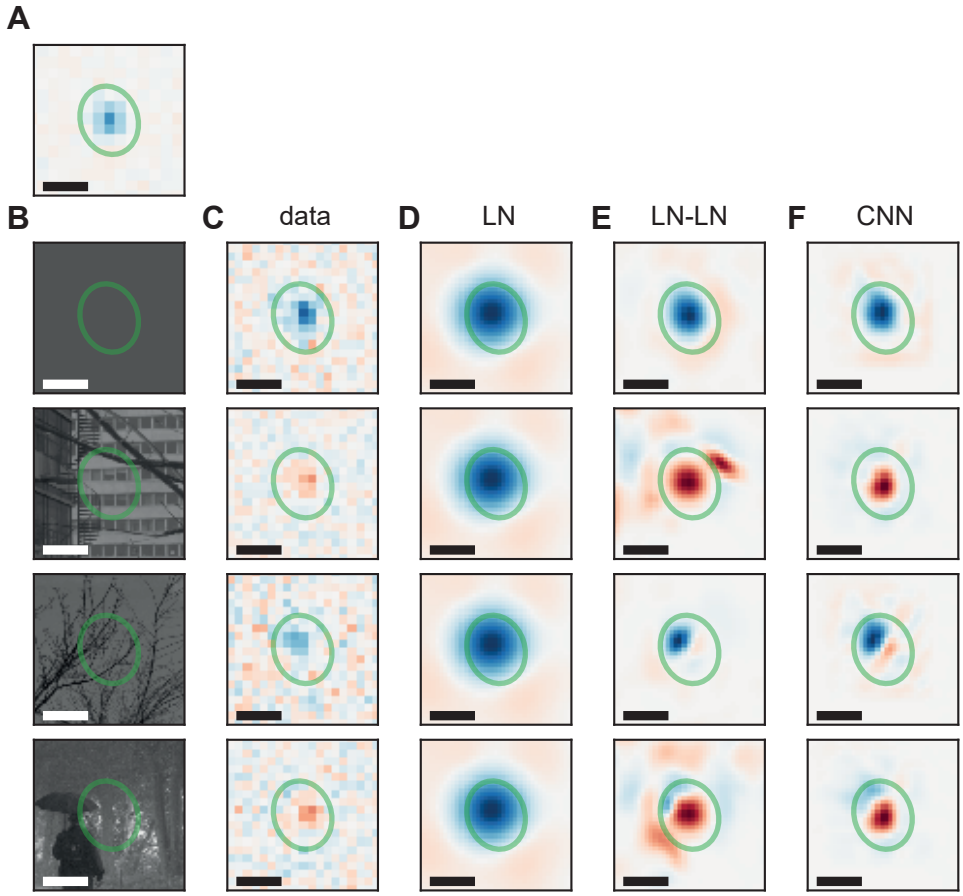

Figure 4.8: LSTAs predicted by the different models (example cell 1). **A.** Spatial STA for this cell. The ellipse (green) approximates its position and shape, and is plotted in the same position in subsequent plots. **B.** Reference images tested as contexts. **C.** LSTAs measured experimentally for each context. **D-F.** Predictions of the LSTA by the LN, LN-LN and CNN models for each context. (scale bars: 300 µm)

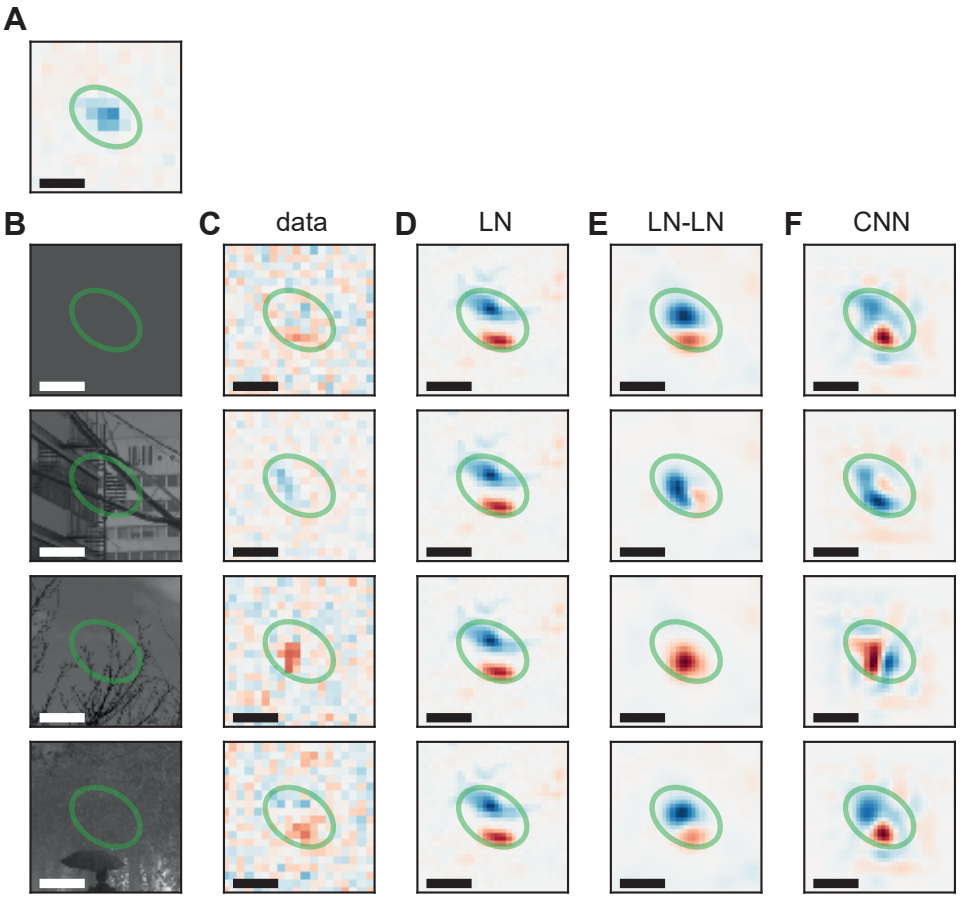

Figure 4.9: LSTA predicted by the different models (example cell 2). Same legend as figure 4.8 for a different ganglion cell.

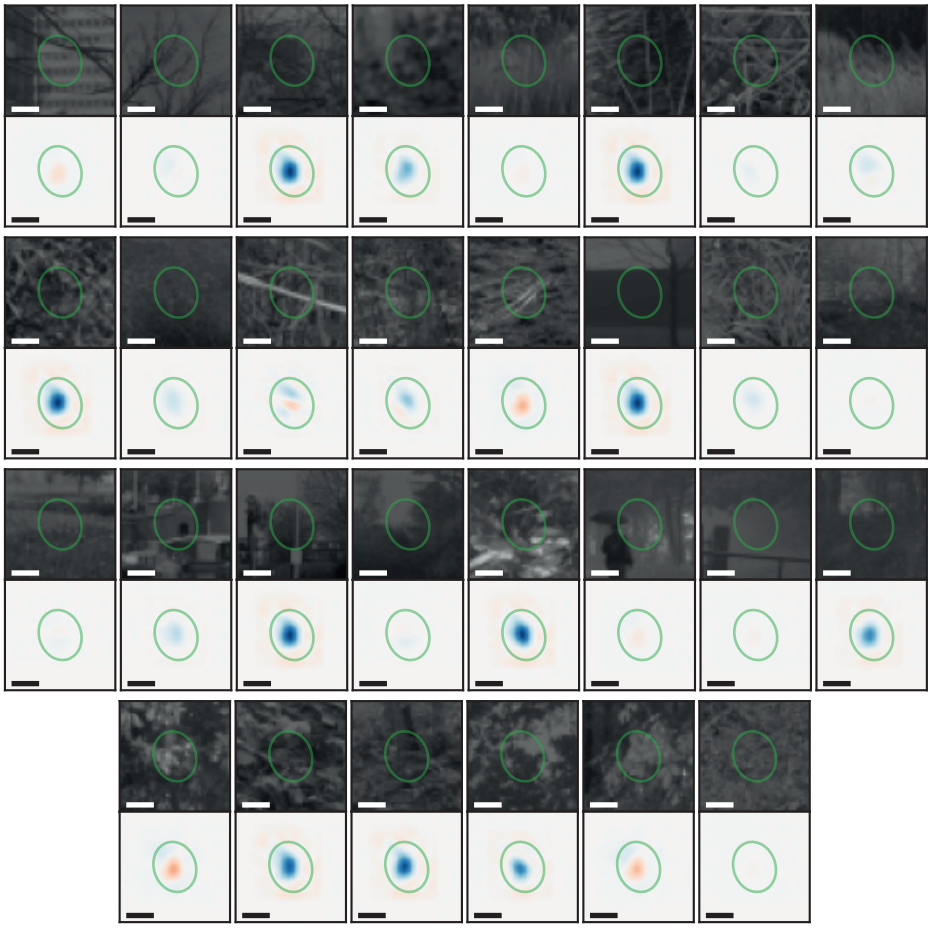

Figure 4.10: Predictions of LSTAs by the CNN model over several contexts of the test set (example cell 1 already shown in figure 4.8). Each natural image is shown above the corresponding LSTA prediction (scale bar: 250 µm). Ellipses (green) represents the position and shape of the spatial STA of the ganglion cell.

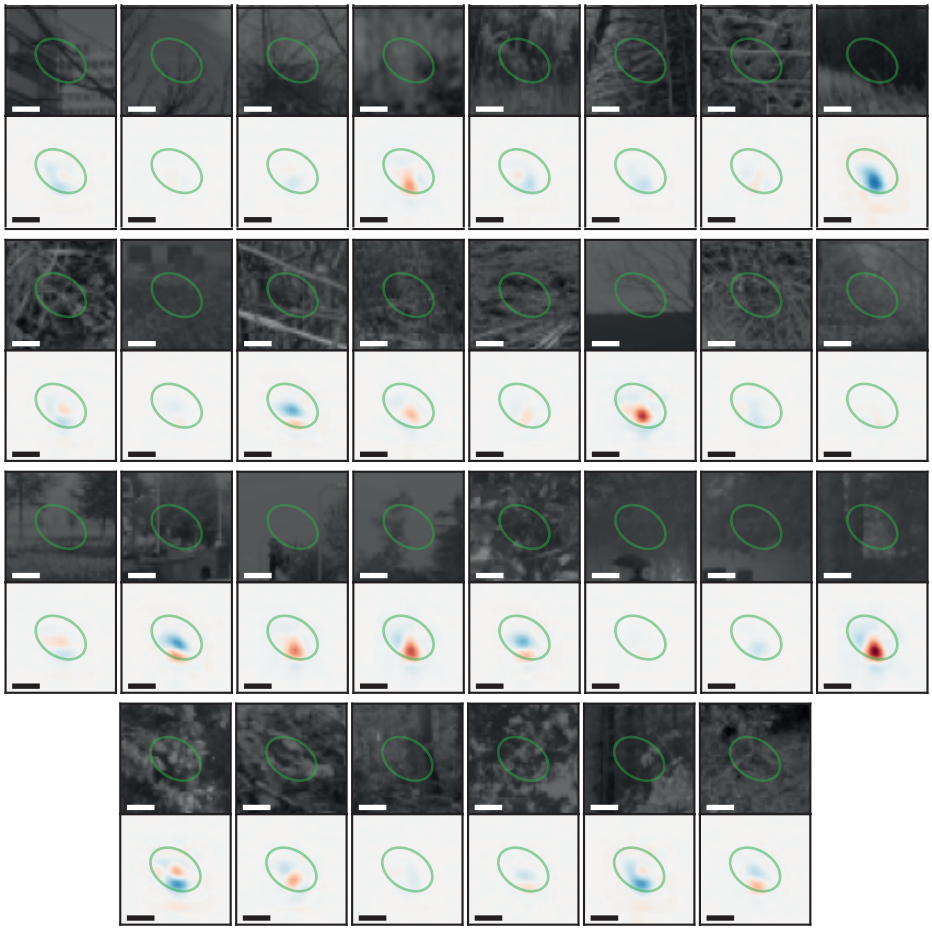

Figure 4.11: Predictions of LSTAs by the CNN model over several contexts of the test set (example cell 2 already shown in figure 4.9). Same legend as figure 4.10 for a different ganglion cell.

in this section are those of the CNN model with only 2 convolutional kernels.

The first components are the convolutional kernels used by the first layer (figure 4.12B). The first convolutional kernel shows a clear center-surround organisation with strong positive weights in its center (red) and weak positive weights in the surround (blue). In the following, we referred to this kernel as the "ON kernel" and its output as the "ON feature map". The second kernel also shows a clear center-surround organisation with negative weights in its center (blue) and a dimmer surround with positive weights (red). Note that this is exactly the opposite of the ON kernel, an "OFF kernel" which produces an "OFF feature map". These observations suggest that the CNN model relies on these two parallel pathways working to make accurate predictions.

The readout weights of the second layer were factorized between the feature weights and the spatial masks, shown in figure 4.12C and 4.12D respectively. The features weights allowed us to distinguish three cell groups. First, some cells have high OFF weight and zero ON weight. They were all classified as OFF cells. Second, one cell has high ON weight and zero OFF weight, and is the only ON cell present in the dataset. Finally, all the other cells that have both a non-zero OFF weight and a non-zero ON weight. For these ganglion cells, the coexistence of the two pathways was necessary to predict their neural responses. The cells for which we observed experimentally different polarities among their LSTAs (highlighted in red) belong to this last group. This suggests that the interplay between an ON pathway and an OFF pathway may underlie the changes in polarity measured among LSTAs.

4.2.8 Putative mechanisms revealed with pharmacology

The ON and OFF kernels could be analogous to the ON and OFF bipolar cell pathway. The model structure suggests that both pathways are necessary to obtain polarity reversals. In preliminary experiments, we tested this hypothesis by measuring LSTAs of ganglion cells before and after applying LAP-4, which blocks selectively the ON pathway. In two example cells, we noticed changes in LSTA polarity that disappeared after drug application (figure 4.13). While these data are preliminary, it suggests that polarity reversal in LSTA require the convergence of the ON and OFF bipolar cell pathways onto the same ganglion cell.

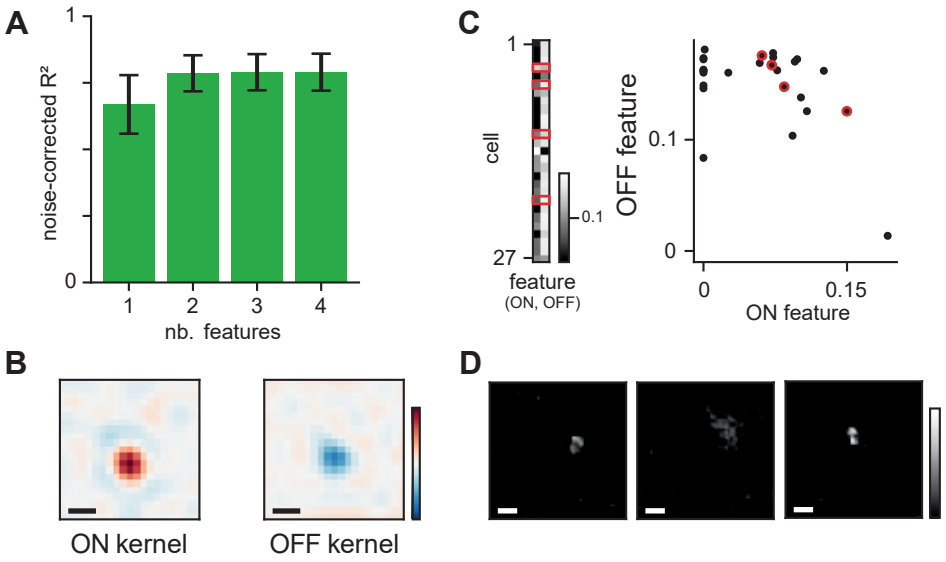

Figure 4.12: Internal components learned by the CNN model. **A.** Effect of the number of features on the performance of the CNN model (error bars: 95 % confidence interval). **B.** Learned convolutional kernels (scale bar: 100 µm, size: 21 × 21 px). The color map is shared by the two kernels from their minimum (blue) to their maximum (red). **C.** Learned feature weights (scale bar: 300 µm). (left) Matrix representation where each row corresponds to a ganglion cell and each column to a feature. (right) Scatter plot where each point corresponds to a ganglion cell. (both) Highlighted elements (surrounded in red) correspond to ganglion cells which clearly presented LSTAs of different polarities. **D.** Learned spatial masks of three example cells (scale bar: 300 µm, size: 88 × 88 px).

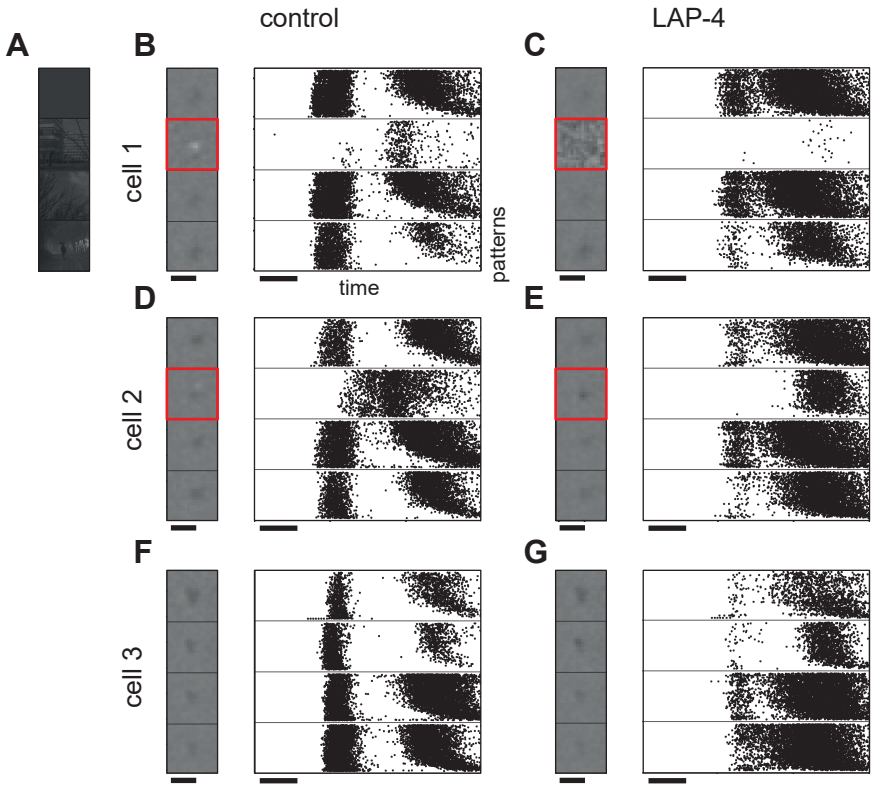

Figure 4.13: Effect of LAP-4 application on LSTAs. **A.** Reference images tested as contexts. **B.** LSTAs (left) and rasters (right) of an example cell before LAP-4 application (scale bars: 500 µm, 50 ms). The highlighted LSTA (surrounded in red) has ON polarity unlike the other LSTAs. For rasters, each plot corresponds to a context, each line to a randomized perturbation pattern, and each point to a spike. **C.** Same as **B.** but with LAP-4 application. The polarity of the highlighted LSTA (surrounded in red) is no longer visible because of the disappearance of responses (right). **D-E.** Same as **B-C.** for another example cell. The highlighted LSTAs (surrounded in red) show variations in shape and polarity. **F-G.** Same as D-E. for a last example cell for which LAP-4 application didn't have any effect.

4.3 Discussion

4.3.1 Variations in LSTAs suggest that ganglion cells code more abstract features like contrast

To better understand these variations in LSTAs with natural images predicted by the CNN model, we performed a principal component analysis (PCA) on the ensemble of LSTAs predicted by the model for all the ∼3000 images used as a stimulus. For most cells, the first two principal components accounted for more than 90% of the variance of these LSTAs. We thus projected all the LSTAs, as well as the natural images themselves, in the 2D space formed by these two first principal components. We represented each image as a point in this 2D space, and the associated LSTA as a vector. We chose this vector field representation because LSTA represents the best change in the reference image to increase the firing rate of the ganglion cell. As such, it represents the gradient of the stimulus-response function. For readability, we binned this 2D space and averaged all the images and corresponding LSTAs that fall in the same bin.

For some ganglion cells this representation shows that the direction of the LSTAs in this 2D space is almost always pointing in the same direction (figure 4.14A). This corresponds to a cell with little variation of the LSTA shape. In this case, an increase of firing rate will always signal a change in the same direction, no matter what the reference image is. This single direction represents a change of the luminance averaged over the receptive field of the cell. As a result, it should be possible to decode local luminance from the activity of this cell.

On the contrary, other ganglion cells show large variation in the direction of the LSTA (figure 4.14B). This corresponds to important changes in the LSTAs, like polarity inversions. In this case, without knowing the reference image, it is impossible to tell if an increase of firing rate corresponds to an increase or a decrease in local luminance. It is thus likely that the activity of this cell contains little information about luminance. However, the vector field representation reveals an interesting structure about the relation between reference image and LSTA shape: there is a region where the LSTA seems to vanish, and in other places, the LSTA point towards a direction that is always opposite to this region. Since LSTA corresponds to a gradient, this corresponds to a local extremum of the stimulus-response function. The single extremum, and the structure of the vector field, is reminiscent of the one produced by a quadratic function. We hypothesized that this vector field could be produced by a cell whose response would be proportional to the contrast inside the receptive field center, where contrast is defined as the

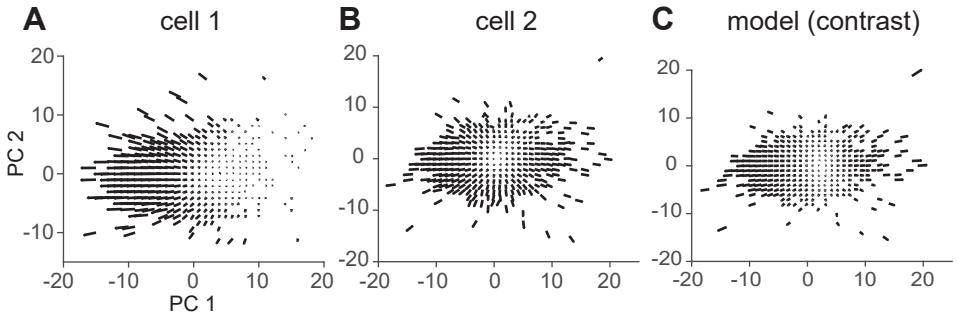

Figure 4.14: Principal components of the LSTAs on the training set. For all subsequent plots, we considered the 2D space formed by the first two principal components of all LSTAs. Each vector correspond to a pair composed of an image and the corresponding LSTA where its origin correspond to the image and its direction and length to the LSTA. For clarity, this 2D space is binned and all the images and corresponding LSTAs that fall in the same bin are averaged. **A.** A first example cell for which all the LSTAs point in the same direction. **B.** A second example cell for which the LSTA directions vary a lot. **C.** Vector field generated from a local contrast function.

square difference between the luminance value and the mean luminance, averaged over the receptive field center. We have generated the same vector field for a local contrast function (see Material and methods, and figure 4.14C). The two vector fields looked qualitatively similar, suggesting that the local contrast could be decoded unambiguously from this cell. While this analysis would need a proper quantification using a mutual information estimation, this vector field representation shows that the variation in LSTA are highly structured. The changes in polarity of the LSTA suggest a very context-dependent code, which is incompatible with a robust encoding of luminance. However, our results suggest that it is compatible with the robust encoding of more abstract features like contrast. Probing LSTAs is thus not only a way to test quantitative models of retinal processing, it can also be a tool to test which features of the image can be robustly coded by a ganglion cell.

4.3.2 Related work

Previous works have shown that OFF ganglion cells can respond selectively to light increase in specific conditions. Tikidji-Hamburyan et al. (2015) showed that an OFF ganglion cell can do so when changing the background luminance by at least one order of magnitude. The change in polarity was due

to the adaptation of the background luminance. Here we have measured the ON/OFF selectivity of ganglion cells for natural images where the overall background luminance was kept constant. Furthermore, in our stimuli all the image presentations, with and without perturbations, were interleaved to avoid any adaptation to the specific statistics of a natural image. The changes in polarity cannot be attributed to adaptation to the background luminance.

Geffen et al. (2007) reported that OFF ganglion cells can switch to ON polarity during and slightly after a sudden shift of a large grating present in the surround. Compared to our study, here the change was linked with a transient and dynamical change in one stimulus property, while we measured the LSTAs for natural images flashed at a constant pace. The results of Geffen et al. (2007) suggest that extending our paradigm to a spatio-temporal case, by applying perturbations on top of a scene with natural dynamics (natural motion, but also saccadic-like movements), could reveal even more complexity in the LSTA dependence on context.

Maheswaranathan, McIntosh, Kastner, et al. (2018) learned a deep network model on retinal responses to natural movies and noticed that the gradient of of the stimulus-response function, estimated with their network model, could change polarity depending on the content of the natural scene. The gradient they calculated is the equivalent of our LSTA. They observed the same context-dependency. The main difference with our approach is that they only measured it on a model, while our estimation of the LSTA allows for a direct experimental measure of how retinal processing can be context-dependent. This LSTA measure can be used to validate models. We observed some models predicted them well, while others did not. It can show a qualitative difference between different model classes, and can highlight the strength or weaknesses that classical testing strategies, where the model performance is quantified by its prediction of the responses to a testing set, could miss. For example, we showed that the prediction performance of LN-LN and CNN models were close, but for some cells LSTA were better predicted by the CNN model.

Beyond its use as a way to validate a model, our perturbative approach is useful to probe selectivity during natural scene stimulation. We have shown how this might help understanding what a given cell is encoding. We assumed that this encoding must be robust to change in the reference image. In our case, this excluded that ganglion cells showing a polarity inversion in the LSTAs could code for local luminance. However, our analyses suggest that they could still code for contrast.

This perturbative approach could also be applied in other sensory systems to refine models or test hypotheses about what features are extracted from

the sensory input. Related works in the visual cortex have learned deep networks online to then find the stimulus that maximized the neuronal response (Bashivan et al., 2019; Ponce et al., 2019). This approach is complementary to ours: while they looked for the "maximizing" stimulus, LSTAs can be seen as the "locally maximizing" stimulus, i.e. the local change that will be best at increasing the response. An interesting outcome of our study is that these local changes do not always point in the same direction. In cells showing polarity inversion, our results suggest that this direction can systematically vary with the reference image. Our vector field representation suggests that this is consistent with the coding of a more non-linear feature, e.g. contrast. Our results thus show that the processing performed by one cell on the visual scene cannot be summarized by one maximizing stimulus. This might suggest that most of the processing can be modeled by projecting the stimulus on a single direction, which corresponds to a "template matching" approach. A more complete understanding can be obtained by measuring many LSTAs, which can be considered as "locally maximizing" stimuli. They show that for many cells, projection on a single direction cannot give a good account of the processing performed by these cells.

Similarly, a related work by Keshishian et al. (2020) in the auditory cortex, using deep network models learned on data, summarized processing by describing 3 different STAs for one cell, and each of them is supposed to tile a different region of the stimulus space. Similarly to (Maheswaranathan, McIntosh, Kastner, et al., 2018), this study did not measure them experimentally. More importantly, our results show that even in the retina, some ganglion cells cannot be reduced to a small number of STAs because they code for a more abstract feature, which results in systematic changes in the LSTA depending on the reference image.

One limitation is that our experimental approach allowed us so far to measure LSTA for only a few reference images. However, strategies have been designed to estimate STA with limited amount of data, making it possible to measure them on more reference images. Applying our online spike sorting might also be a way to test more points using active learning strategies, where the stimulus is modified depending on the previous responses, to optimize the exploration of the stimulus space.

4.4 Material and Methods

4.4.1 Electrophysiology

All the electrophysiological data were obtained from an isolated retina from an adult axolotl salamander (*Ambystoma mexicanum*, pigmented wild-type). The experiment was performed in accordance with institutional animal care standards of Sorbonne Université. After killing the animal, the eye was enucleated and dissected under dim light condition. We hemisected the eye along the edge of the cornea. We removed the vitreous humor. We separated the retina from the eyecup. We removed the pigment epithelium. We cut it in half. We strechted beforehand a perforated dialysis membrane over a circular plastic holder. We mounted one of the retina halves onto the membrane such that the photoreceptors were facing the membrane. We manipulated the holder to position the retina onto a microelectrode array (MEA) so that the retinal ganglion cell layer laid over the electrodes of the MEA. During dissection and recordings, the tissue was perfused with oxygenated Ames solution and a peristaltic perfusion system with 2 independent pumps: PPS2 (Multichannel Systems GmbH). The retina was kept at room temperature during the whole experiment.

4.4.2 Data acquisition

The data acquisition system (from Multichannel Systems GmbH) included a microelectrode array (256MEA30/8iR-ITO), an amplifier and data acquisition hardware (USB-MEA256-System) and a data acquisition software (MC_Rack). The MEA had 256 circular electrodes arranged in 16×16 grid layout with 8 µm diameters and 30 µm inter-electrode distances. Extracellular voltage signals of retinal ganglion cells were recorded from 252 active electrodes. The data sampling rate was 20 kHz per electrode. We high-pass filtered the raw data at 200 Hz, and isolated the spikes using SpyKING CIRCUS (Yger et al., 2018). Subsequent data analysis were done with custom-made Python codes.

We extracted the activity of a total of 27 neurons for a single experiment which satisfied multiple constraints. We kept cells with a low number of refractory period violations and whose template waveform could be well distinguished from the template waveforms of other cells. These constraints ensured a good quality of the reconstructed spike trains. In addition, we discarded neurons that show no or almost no responses to flashed images preventing the estimation of LSTAs.

4.4.3 Visual stimulation

We used a white mounted LED (MCWHLP1, Thorlabs Inc.) as a light source, and controlled the light intensity with a LED driver (LEDD1B, Thorlabs Inc.). We used a dim mean light level which corresponds to mesopic vision. The stimuli were displayed using a Digital Mirror Device (V-Module, ViALUX GmbH) and focused on the outermost layer of the retina, where the photoreceptors are localised, using standard optics. We displayed a random binary checkerboard during 1 hour at 30 Hz to map the receptive fields of ganglion cells (check size: 73.5 µm, checkerboard size: 50×50 checks). All the other stimuli are described in subsequent method sections.

4.4.4 Natural image stimuli

We used the van Hateren's Natural Image Dataset (Van Hateren and Schaaf, 1998). This dataset consists of 4212 monochrome and calibrated images taken in various natural environment (wood, open landscapes and urban area). The calibration ensure that there is a strictly linear relationship between scene luminance and pixel value. These images are known to be slightly blurred by the point-spread function of the camera (optics of the lens). We didn't discard any images based on this criterion. It is also known that the global statistics of these images (considered sequentially) may change regularly since they were taken in typical series of 100-200 consecutive images where each series correspond to a different natural environment. This aspect was more problematic because the retinal system is known to adapt its sensitivity to the range of light intensities encountered in different environments.

To avoid that these adjustments alters the rules by which a neuron responds to sensory input, we devised a preprocessing step to overcome this problem. First, we identified the images with a significant number of pixels which have reached saturation. To do so we defined a saturation level as the proportion of pixels above a given threshold value (6266 for ISO 200, 12551 for ISO 400 and 25102 for ISO 800, these values where found manually by inspection of the image histograms). If the saturation level was above 2% then we decided to discard the image. The discarded images were found to present surfaces with high albedo (e.g. snow, clouds). This resulted with a total of 3190 images with acceptable saturation level. Second, we cropped the central part of each image (initial size: 1536×1024 px, final size: 864×864 px) to match the targeted field of view (3.024×3.024 mm^2 with a pixel projection of 3.5 µm). Third, for each image, we converted the pixel values to luminance with the conversion factor provided by the calibration. Fourth, we normalized the images with a custom procedure to fix the mean luminance

and the root mean square (RMS) contrast. We replaced the linear scale by a log scale. We centered and scaled the distribution of pixel values. We came back to the linear scale where we centered and scaled the distribution of pixel values for a second time to fix the mean luminance to 0.25 and the standard deviation to 0.05. We clipped the pixel values below 0 to 0, and those above 1 to 1.

Unperturbed natural image stimulus The dataset contains the 3190 images obtained after the preprocessing step described in the previous paragraph. This dataset has been used to create multiple models of the response of ganglion cells to natural images. We distinguished 30 images which were presented multiple times (20 repetitions each) to create the test dataset. 10 % of the remaining images were allocated to the validation dataset while the rest of the images composed the training dataset.

Perturbed natural image stimulus The generation of the perturbed natural image stimulus consisted in superimposing reference images with multiple perturbation patterns. We choose 3 reference images among the preprocessed Van Hateren images. We set the perturbation patterns as checkerboard patterns without spatial structure (i.e. uncorrelated). Each check (i.e. spatial location) is associated with a random and independent value, either -1 or $+1$ (with equal probability). Given a perturbation amplitude, these perturbation patterns are scaled and then independently added to the reference image to result with the grayscale images to be flashed. We used a check size of 52.5 µm and the checkerboard had 56×56 checks (i.e. 2.94×2.94 mm^2). The perturbed area was slightly smaller than the targeted field of view, and was added at its center. We used a perturbation amplitude of 0.03125 in normalized luminance which correspond to a perturbation amplitude of 8 gray levels using a 8-bit grayscale. This amplitude has been chosen based on preliminary experiments with amplitude ranging from 2 to 30 gray levels have been tested (data not shown).

4.4.5 Linear-nonlinear model (LN)

We implemented a regularized LN model. This model is fitted for each neuron separately and consists of: a) linear filter weights $w_{i,j}$ that compute a dot product with the input images (where i and j index space), b) a pointwise nonlinear function f_θ that converts the filter output into a non-negative spike rate, and c) a Poisson noise model for training (since spike counts are assumed to be generated through a Poisson process). We chose f_θ to be a parametric

softplus function such that:

$$f_\theta(x) = \alpha \ln\Big(1 + \exp\big(\beta x + \gamma\big)\Big) \qquad \text{where } \theta = \{\alpha, \beta, \gamma\} \tag{4.1}$$

The spiking rate r of a neuron given an input image $\mathbf{X}$ will follow:

$$r(\mathbf{X}) = f_\theta\left(\sum_{i,j} w_{i,j} x_{i,j}\right) \qquad \text{where } \mathbf{X} = (x_{i,j}) \tag{4.2}$$

Regularization Additionally, two types of regularization were applied:

- L_1 regularization to induce sparsity since we expected the spatial pooling to be localized:

$$\mathcal{L}_{L_1} = \sum_{i,j} |w_{i,j}| \tag{4.3}$$

- Laplacian regularization to induce smoothness since we expected the spatial pooling to be localized on contiguous pixels:

$$\mathcal{L}_\Delta = \sqrt{\sum_{i,j} (\mathbf{W} * \mathbf{\Delta})^2_{i,j}} \qquad \text{where } \mathbf{W} = (w_{i,j}) \text{ and } \mathbf{\Delta} = \frac{1}{4}\begin{bmatrix} 1 & 2 & 1 \\ 2 & -12 & 2 \\ 1 & 2 & 1 \end{bmatrix} \tag{4.4}$$

  with zero padding such that $\mathbf{W} * \mathbf{\Delta}$ had the same spatial dimensions as $\mathbf{W}$.

Constraint Moreover the following constraints were applied: a) force weights to have at most unit norm, b) force α to be non-negative (to guarantee non-negative firing rates), and c) force β to be non-negative (to guarantee that neuron polarity is only captured by the filter).

Initialization To initialize the weights, we took the "normalized" spike-triggered average (STA) for each neuron as our initial guess and added, for each weight, a random component sampled from a truncated normal distribution with mean zero and standard deviation 0.01. It is supposed to help and accelerate the convergence to the trained model. The initialization of the nonlinear function was identical for all neurons with α set to 1, β set to 1 and γ set to 0.

4.4.6 Linear-nonlinear linear-nonlinear model (LN-LN)

This model was also fitted for each neuron separately. It is composed of two layers. Each layer performs a linear combination of its inputs followed by a nonlinear transformation. We can think of the first layer as a collection of two LN units which are convolved in space. Thus the output of the first layer corresponds to two feature maps, one for each unit. The second layer is a single LN unit taking this output as input.

More precisely, the model consists of: a) convolutional kernel weights $k^{\text{on}}_{i,j}$ and $k^{\text{off}}_{i,j}$ that compute two convolutions with the input images (where i and j index space), b) two pointwise nonlinear function $f_{\theta^{\text{on}}}$ and $f_{\theta^{\text{off}}}$ that converts the convolutional outputs into non-negative activation values, c) linear filter weights $w^{\text{on}}_{i,j}$ and $w^{\text{off}}_{i,j}$ that compute a dot product with the activation maps, d) a pointwise nonlinear function f_θ that converts the filter output into a non-negative spike rate, and e) a Poisson noise model for training. The pointwise nonlinear functions were chosen to be parametric softplus functions with possibly different value for their parameters θ^{on}, θ^{off} and θ.

The outputs of the ON unit and the OFF unit of the first layer were therefore:

$$\mathbf{A}^{\text{on}} = f_{\theta^{\text{on}}}\left(\mathbf{K}^{\text{on}} * \mathbf{X}\right) \qquad \text{and} \qquad \mathbf{A}^{\text{off}} = f_{\theta^{\text{off}}}\left(\mathbf{K}^{\text{off}} * \mathbf{X}\right) \tag{4.5}$$

Thus the spiking rate r of a neuron given an input image $\mathbf{X}$ was:

$$r(\mathbf{X}) = f_\theta\left(\sum_{i,j} w^{\text{on}}_{i,j} a^{\text{on}}_{i,j} + \sum_{i,j} w^{\text{off}}_{i,j} a^{\text{off}}_{i,j}\right) \tag{4.6}$$

Regularization We used the same kinds of regularization as those used for the LN model. The L_1 and Laplacian regularizations were only applied on the linear filters of the second layer since the kernels of the first layer were kept constant. The Laplacian regularization was adapted to apply to the linear filter weights for each feature:

$$\mathcal{L}_\Delta = \sqrt{\sum_{i,j}(\mathbf{W}^{\text{on}} * \boldsymbol{\Delta})^2_{i,j} + \sum_{i,j}(\mathbf{W}^{\text{off}} * \boldsymbol{\Delta})^2_{i,j}} \tag{4.7}$$

where $\mathbf{W}^{\text{on}} = \left(w^{\text{on}}_{i,j}\right)$, $\mathbf{W}^{\text{off}} = \left(w^{\text{off}}_{i,j}\right)$, and $\boldsymbol{\Delta}$ is the discrete Laplace operator already described in equation 4.4.

Constraints We used the same constraints as those used for the LN model. The max norm constraint was applied independently to $\mathbf{W}^{\text{on}}$ and to $\mathbf{W}^{\text{off}}$.

Initialization To initialize the convolutional kernels, we used 2D Gaussian spatial profiles of opposite polarity but sharing the same spatial scale such that:

$$\mathbf{K}^{\text{on}}_{i,j} = \frac{1}{2\pi\sigma^2} e^{-\frac{\Delta^2\left((i-i_0)^2+(j-j_0)^2\right)}{2\sigma^2}} \qquad \text{and} \qquad \mathbf{K}^{\text{off}}_{i,j} = -\mathbf{K}^{\text{on}}_{i,j} \tag{4.8}$$

where $\sigma = \frac{s}{\Delta}$ with $s = 30$ µm the targeted spatial scale and $\Delta = 28$ µm the size of one pixel (after downscaling, i.e. 8×3.5 µm). The spatial scale was chosen to match theoretically this profile with the profile of a typical bipolar receptive field found in the salamander retina. We initialized the linear filters with weights sampled from a truncated normal distribution with mean 0 and standard deviation 0.01. For this model, we didn't use an initial guess related to the "normalized" spike triggered average (STA) to try to speed up the convergence during the training procedure. About the pointwise nonlinear functions, theirs initializations were identical to the one described for the LN model.

4.4.7 Convolutional neural network model (CNN)

The design of this model has been inspired by previous work. First, by the deep convolutional neural network model described in (McIntosh et al., 2016; Maheswaranathan, McIntosh, Kastner, et al., 2018; Maheswaranathan, McIntosh, Tanaka, et al., 2019) where authors show that this kind of model can capture retinal responses to natural scene stimuli. Second, by the novel architecture of deep neural networks described in (Klindt et al., 2017; Cadena et al., 2019) where a deep convolutional neural network is meant to learn a convolutional feature space shared across cells, and where a sparse readout layer (which factorizes the spatial and feature dimensions) allows for separating the inference problem of each cell's receptive field location and type from the inference of the feature space.

This model is rather different from the previous ones since its training phase doesn't consider each cell one by one but all the cells at once instead. It is composed of two layers. Multiple LN units are convolved in space in the first layer resulting in multiple activation maps. Then the second layer implements the readout mechanism. For each cell, the activation maps are filtered by readout weights which are specific to this cell and factorized in space and feature. Finally, the resulting activation value passes through the final nonlinear activation function associated to this cell to give the predicted spiking rate.

More precisely, the model consist of: a) convolutional kernel weights $k_{i,j,k}$ that compute convolutions with the input images (where i and j index space

and k index kernels), b) pointwise nonlinear functions $f_{\theta_k^{[1]}}$ that convert the convolutional outputs into non-negative activation values. And in addition for each neuron n: c) readout weights $w_{i,j,k,n}$ which can be factorized as $w_{i,j,k,n} = u_{i,j,n}v_{k,n}$ where $u_{i,j,n}$ represent the spatial weights and $v_{k,n}$ the feature weights, d) a pointwise nonlinear function $f_{\theta_n^{[2]}}$, and e) a Poisson noise model for training. We choose $f_{\theta_k^{[1]}}$ and $f_{\theta_n^{[2]}}$ to be parametric softplus functions with possibly different value for their parameters.

The outputs of the kth unit of the first layer were:

$$\mathbf{A}_{:,:,k} = f_{\theta_k^{[1]}}(\mathbf{K}_{:,:,k} * \mathbf{X}) \tag{4.9}$$

Such that the spiking rate r_n of the nth neuron given an input image $\mathbf{X}$ was:

$$r_n(\mathbf{X}) = f_{\theta_n^{[2]}}\left(\sum_k \sum_{i,j} u_{i,j,n}v_{k,n}a_{i,j,k}\right) \tag{4.10}$$

Additionally, batch normalization was applied to the outputs of the first layer for stability of the learning phase.

Regularization We used a Laplacian regularization on the convolutional kernels of the first layer, and we used L_1 regularization on the spatial weights and on the feature weights of the second layer such that:

$$\mathcal{L}_\Delta = \sqrt{\sum_k \sum_{i,j} (\mathbf{K}_{:,:,k} * \mathbf{\Delta})^2_{i,j}} \qquad \text{and} \qquad \mathcal{L}_{L_1} = \lambda_s \sum_{i,j,n} |u_{i,j,n}| \; + \; \lambda_f \sum_{k,n} |v_{k,n}| \tag{4.11}$$

Constraints We constrained the spatial weights and the features weights to be positive. This constraint was applied independently on $\mathbf{U}$ and $\mathbf{V}$. We initialized the linear filters

Initialization To initialize the convolutional kernels, we initialized each weight with a truncated normal distribution with mean 0 and standard deviation 0.01. The spatial weights were initialized with the "normalized" spatial spike triggered average (STA) of each neuron (with their amplitude set to 0.1) corrupted with a small Gaussian noise (with mean 0 and standard deviation 0.01). The feature weights were initialized with a truncated normal distribution with mean 0 and standard deviation 0.01. This initialization is supposed to speed up the convergence during the training procedure. The pointwise nonlinear functions were initialized in the same way as before.

4.4.8 Model training and cross-validation

Model fitting Considering the n recorded image-response pairs $(\mathbf{X}_1, y_1), \ldots, (\mathbf{X}_n, y_n)$ for one neuron, the resulting loss function is given by:

$$\mathcal{L} = \left(\frac{1}{n} \sum_{k=1}^{n} r\left(\mathbf{X}_k\right) - y_k \log r\left(\mathbf{X}_k\right) \right) + \lambda_{L_1} \mathcal{L}_{L_1} + \lambda_{\Delta} \mathcal{L}_{\Delta} \tag{4.12}$$

where the first term corresponds to the negative log-likelihood of the Poisson loss and where λ_{L_1} and λ_{Δ} are the hyperparameters which controls the importance of the regularization terms.

We fitted the model by minimizing this loss using the Adam optimizer on the training set. The training set was composed of 2910 natural images and the associated responses (i.e. number of spikes elicited between the onset and the offset of the flash). We fixed the batch size to 64. We used a learning rate initially equal to 0.002 and added a decay of the learning rate, when 10 consecutive training epochs did not improve the validation loss we came back to the first epochs and divided the learning rate by 4 a factor of two. In addition, we used early stopping with a patience of 15 training epochs. These two mechanisms prevented overfitting.

We cross-validated the hyperparameters λ_{L_1} and λ_{Δ} for each neuron independently by performing a random search. For each hyperparameter λ, we used 128 values sampled from the corresponding continuous random variable Λ defined over $[\lambda_{min}, \lambda_{max}]$ such that $\log(\Lambda) \sim U\left[\log(\lambda_{min}), \log(\lambda_{max})\right]$. For λ_{L_1} we usually searched over $[1e-6, 1e-1]$ and for λ_{Δ} we searched over $[1e-6, 1e+2]$.

When fitting models, we used the same split of data for training, validation, and testing across models.

4.4.9 Model evaluation

To evaluate the performance of the models, we used a testing set of 30 different stimuli where each stimulus have been repeated 20 times. These repetitions allowed to separate prediction error in two parts. The error due to the limitations of the model and the error due to the intrinsic noise in the response.

Given n responses to the same stimulus, $y_1, \cdots, y_n$, we averaged them over odd and even numbered trials to get estimates of the actual mean response, $\bar{y}_o$ and $\bar{y}_e$. We defined the reliability as the correlation between these estimates, $r_{\bar{y}_o, \bar{y}_e}$. Then, given the prediction of one model, $\hat{y}$, we defined the accuracy as the correlation between the prediction and one the two estimates of the mean response, $r_{\hat{y}, \bar{y}_o}$.

Without noise correction, we reported the R-squared of the correlation, $R^2 := r^2_{\hat{y},\bar{y}_o}$. Following the methods of (Schoppe et al., 2016; Hsu et al., 2004), we estimated the maximum correlation attainable by a model:

$$r_{max} := \sqrt{\frac{2}{1 + \sqrt{\frac{1}{r^2_{\bar{y}_o,\bar{y}_e}}}}} \tag{4.13}$$

and reported the corresponding upper bound, $R^2_{max} := r^2_{max}$.

With noise correction, we estimated the noise-corrected correlation:

$$r_{nc} := \frac{\frac{1}{2}\left(r_{\hat{y},\bar{y}_o} + r_{\hat{y},\bar{y}_e}\right)}{\sqrt{r_{\bar{y}_o,\bar{y}_e}}} \tag{4.14}$$

similar to the methods used by (Keshishian et al., 2020). We reported the corresponding noise-corrected R-squared, $R^2_{nc} := r^2_{nc}$.

4.4.10 LSTA prediction

Given a model which predicts the firing rate $r(\mathbf{X})$ of a neuron in response to image $\mathbf{X}$, we predicted the LSTA with:

$$\text{LSTA}(\mathbf{X}) := \frac{\partial r(\mathbf{X})}{\partial \mathbf{X}} = \left(\frac{\partial r(\mathbf{X})}{\partial x_{i,j}}\right) \tag{4.15}$$

the gradient of the model output with respect to the input image. This quantity is also referred as the data Jacobian matrix (S. Wang et al., 2016). Computing the quantity is particularly straightforward for deep convolutional neural networks. These networks can be viewed as a sequential composition of convolutions and nonlinear functions. The gradients of these components taken individually can be easily calculated. Then, the application of the chain rule allows to compute the gradients of the composite function. Modern machine learning libraries such as TensorFlow (Abadi et al., 2016) or PyTorch (Paszke, Gross, Massa, et al., 2019; Paszke, Gross, Chintala, et al., 2017) provide automatic differentiation systems which allows to avoid the manual calculation of these gradients. Since all our models were implemented with TensorFlow, we took advantage of automatic differentiation to calculate the LSTAs.

One issue with this method is that the gradient is equal to the LSTA up to a constant multiplier. This constant is determined by the amplitude of the perturbations used to measure the LSTA and by the standardization of images used to train the models. In future work we need to estimate this constant to make a fully quantitative comparison.

Discussion

Modern electrophysiology

Recording techniques to access the spiking activity of large populations of neurons greatly improved over the last decades. Although the work presented in this thesis mainly focused on dense and large-scale microelectrode arrays, this is not the only way of measuring neuronal activity. Imaging techniques are the main alternative to electrical techniques. They rely on fluorescent markers placed in neurons whose light emissions are linked to their electrical variations. Their emitted light can be imaged with an optical microscope to give an indirect measure of the spiking activity. Two-photon calcium imaging is probably the most widely used imaging technique (Garaschuk et al., 2000; Stosiek et al., 2003). Alternatively, voltage sensitive dyes (VSDs) have been developed (Grinvald and Hildesheim, 2004). In both cases, tens of thousands of cells can be recorded simultaneously. However, the temporal resolution is limited and doesn't generally allow for resolving individual action potentials (Theis et al., 2016). Recently, genetically-encoded voltage indicators (GEVIs) appeared as a promising solution to this kinetic problem (Lin and Schnitzer, 2016; Chamberland et al., 2017).

There are some limitations to imaging techniques over electrical techniques. A first limitation is light scattering through biological tissues which imposes a depth constraint. Even if multi-photon fluorescent microscopy can partly relax this constraint, deep brain structures remains inaccessible for imaging (Zipfel et al., 2003). Another limitation is that the microscope systems used to image the fluorescent activity are not portable. As a consequence, animals have their head fixed during the imaging and visual and auditory stimuli are used to recreate virtual environments. Thus, these conditions are not suitable for chronic recordings of freely moving animals which are of particular interest for behavioral studies. Head-mounted and miniature microscopes have been constructed but without comparable capabilities (Cai et al., 2016). This is not the case for brain implants such as silicon probes which are of particular interest for medical purposes like the restoration of

some brain functions after a stroke or other head injuries. Thus, the recent development of imaging techniques is unlikely to supplant the electrical techniques. Instead they will certainly both provide an unprecedented amount of recorded data from various neural systems which will help understand the governing principles and mechanisms of the brain. There will be an increased need for an accurate, efficient and automated spike sorting procedure that scales with the number of microelectrodes.

Spike sorting of non-stationary data

One of the main assumptions shared by spike sorting algorithms is the stationarity of the data. When a neuron elicit a spike, it is supposed to evoke the same spatiotemporal voltage pattern (i.e. template) at the tips of the microelectrodes. However, this assumption doesn't always hold. In the case of the retina, the tissue can slightly relax over time and there is a progressive and irremediable degeneration of cells for recording of several hours (more than 4 hours). Some ganglion cells drift weakly in space which change their electrical footprints seen by the recording device. For examples, if the cell drifts a bit away from the microelectrodes then the amplitude of its template decreases, and if the cell drifts a bit along the microelectrodes then the template can be shifted over the electrodes.

One solution mentioned in this thesis to deal with this kind of drift is to use a template matching procedure where templates can be scaled in amplitudes to some extent. If the scaling is made time dependent then the activity of the cells which are drifting away can be better reconstructed. Another solution has been proposed with the development of the online spike sorting. Using an online clustering algorithm, the dictionary of templates can be updated continuously as spikes are collected through the experiment. If a cell drifts slowly then the corresponding template will slowly change accordingly. However, it is important to note that this method has not been extensively validated yet. Ground-truth dataset of ganglion cells used to validated offline spike sorting algorithms are often too short to show challenging drifts (less than 20 minutes). The generation of synthetic dataset with simulated drifts could compensate this lack with a careful design to keep realistic scenarios. Therefore, a future effort will tackle the extensive validation of the tracking of drifts.

It is more difficult to handle drifts when silicon probes are used to record neurons in the cortex. The brain floats inside the skull and any movement from the animal make it move. These movements can be fast and provoke brief drifts of tens of micrometers which add up with slow drifts. Fast drifts are not a problem if the animal is not behaving but most neuroscience exper-

iments typically involve some motor actions to perform a task. The approach mentioned previously to resolve drifts is not sufficiently mature to be used in this context. Further research and development will be required to enforce robustness to those fast and transient non-stationarities.

A more important and less explored source of non-stationarity is due to the use of pharmacology. In this work, L-2-amino-4-phosphonobutyric acid (L-AP4) have been used to eliminate light response of ON bipolar cells. For some cells, the spike sorting algorithm identifies a template for the spikes elicited before the application of the pharmacology and another template during its application. Even if this kind of pair of templates can be merged during the manual curation, a dedicated procedure would be required to cope with this non-stationary process and to have a fully automated spike sorting procedure in this specific condition.

Adaptive sampling with online spike sorting

Neurophysiological experiments often seek to unravel the function of a biophysical system seen as an input-output system. For example in the retina, the inputs are the spatiotemporal patterns of light which enter the eye while the spike trains of retinal ganglion cells are the outputs. The experimental time is often limited and constrains the number of test inputs that can be used to probe the system. Usually a predefined set of inputs or inputs drawn from a predefined probability distribution are used. In the end, the amount of information provided by some inputs on the system is lower than others. For instance, ganglion cells may not elicit any response to stimuli belonging to a specific region of the stimulus space. In this case, a significant amount of time is wasted to test not very informative inputs.

Instead, given the ability to measure the output during the time course of the experiment, one could start with a small number of initial inputs to acquire the first data about the system. Once some initial information have been obtained, the choice of the inputs can be reconsidered to focus on the most useful to test, and that depends on the objective of the experimenter. Initially, the mutual information between the stimulus and the response has been proposed as an optimal criterion (Machens, 2002). This method has been successfully applied on single-unit recordings of locust and grasshopper auditory receptor neurons (Edin et al., 2004; Machens et al., 2005). This technique allows to find optimal stimulus ensembles under specific constraints.

Another possible objective is to learn a model adaptively. Given a parameterized set of models, the optimal criterion is based on the mutual information between stimulus-response pair and the parameters of the model (Paninski, 2005). However, this Bayesian and information-theoretic approach

needs some adaptations to scale well in high dimensions which is critical for real-time applications (Lewi et al., 2009). In the case of receptive field estimation, the exploitation of structured prior knowledge such as the smoothness and sparsity of receptive fields in space and time provides additional speed up (M. Park and Pillow, 2012). This last method has been used successfully to estimate the receptive fields of simple cell recorded in primate V1 for a simulated experiment, i.e. offline analysis of real neural data (M. Park and Pillow, 2012).

So far, all the techniques mentioned previously relied on single cell recordings but the same adaptive methods can be applied to populations of neurons in live experiments. In this case, objective functions need to be rephrased to apply to ensembles of cells, and result in different stimulus selection. A first example is to choose stimuli that maximize the responses of a specific neuron in the population, which is not that different from a single cell method. Another possibility is to maximize the responses of all the neurons. A last example is to choose stimuli that maximize the scatter of the population responses. All these three objective functions have been considered by Cowley et al. (2017) and used for adaptive sampling of populations of V4 neurons (rhesus macaques). This results in a principled way to find a set of natural images which evoke higher mean responses and elicit a larger diversity of responses.

Recently, this approach gained a particular interest to investigate the feature selectivity of neural systems about which we understand much less such as the visual area 4 (V4). Multiple systems have been proposed to generate natural images rather than selecting them from a pre-existing pool of candidate images such as Cowley et al. (2017). These systems relies on models of the statistics of natural images which allow to represent a larger image space, and make it possible not to constrain the adaptive search. The addition of these generative systems was key to find the optimal images with respect to the objective functions. For example, Ponce et al. (2019) investigated neuronal selectivity in the inferotemporal cortex of rhesus macaques using chronic microelectrode arrays. Bashivan et al. (2019) manipulated the spiking activity of populations of V4 neurons also recorded in the visual cortex of rhesus macaques with microelectrode arrays. In particular, in addition to driving the response of a targeted neural unit to its maximum, they simultaneously kept the responses of all other units low. This results in the generation of images with specific features which are supposed to be those to which V4 neurons are selective. Walker et al. (2019) used a similar approach for V1 neurons recorded in mice using two-photon imaging.

This kind of investigations which use chronic micro-electrode arrays usually rely on multi-unit activity rather than single-unit activity. They don't

drive the responses of single neuron but of small group of neurons. These neurons might be of different type and thus selective to different visual features. In this case, pooling the activity of multiple cells might obfuscate their selectivity. The online spike sorting framework presented in this thesis can solve this issue by providing the activity of individual neurons.

Further research could use this approach for retinal ganglion cells, which show a high diversity of cell types. The simplest objective function to test would be the maximization of the response of a single cell. Since Walker et al. (2019) showed that the most exciting inputs of V1 neurons deviate substantially from the classical notion of Gabor-shaped receptive fields, it would be particularly interesting to see if stimuli that maximally drive ganglion cells deviate substantially from the antagonistic receptive field structure usually found in the retina. At the same time, the results presented in the last part of this thesis show that the stimulus that maximally drive a ganglion cell is a moving target, and depends on the visual context. Thus, the choice of the objective function should be reconsidered to integrate this contextual aspect. With this modification, one may find a good agreement between the optimal stimulus ensemble and the expected shape of the receptive fields in the retina or in V1.

Another application of adaptive sampling with online spike sorting concerns the measurement of LSTAs. Currently, the main limitation is the small number of natural contexts that can be explored during one experiment. Adaptive sampling associated to a tailored objective function could be used to speed up the estimation of LSTAs similar to what have been done for the receptive field estimation with an information-theoretic approach mentioned previously (M. Park and Pillow, 2012). This method would significatively increased the number of natural images that could be tested as natural contexts in a single experiment. It would also make possible a more extensive validation of the prediction of LSTAs by the CNN model.

Another issue with the experimental observation of LSTAs is that a ganglion cell can elicit no spike at all for a given natural image. This means that the activation of the cell is completely rectified for this specific stimulus and that any small perturbation of the stimulus is likely to evoke the same lack of response. In this case it might not be worth spending too much experimental time measuring the corresponding LSTA. In other words, online spike sorting could be used to explore multiple natural images and choose those which evoke responses in most ganglion cells. This strategy would make it possible to then use perturbed natural images to measure LSTAs among a large fraction of the population.

LSTAs as a model free approach to neural computations

In this thesis a new perturbative approach has been introduced to probe selectivity of retinal ganglion cells during natural scene stimulation. This approach finds the perturbed stimulus that locally maximizes the neuronal responses. Interestingly, the perturbations found are different from one image to the other. Even if the actual number of natural images that can be tested experimentally is limited, further optimizations of the experimental protocol should be able to increase significantly this number. In this case, the ability to measure many locally maximizing perturbations would lead to a model-free characterization of the processing performed by the cell. The use of a convolutional neural network model as a proxy for measuring LSTAs might no longer be necessary and direct measurements could be used to test which features of natural images can be robustly coded by retinal ganglion cells.

Instead of using flashed natural images, one could think of using stimuli with temporal dynamics such as movie clips of natural scenes. However, this kind of stimulus would increase the dimension of the perturbation space and the number of perturbed stimuli needed to accurately estimate the LSTAs. Even with optimizations such as adaptive sampling of the perturbations, it is not clear if LSTAs can be measured without a model given the limited experimental time.

Conclusion

We have presented different approaches to reconstruct the activity of populations of neurons at a single cell resolution from extracellular voltage recordings. Template matching based methods are particularly efficient to deal with overlapping spikes which are not rare events since synchronous activity is often observed between nearby neurons. Distributed computing is another key aspect to scale with the ever increasing number of microelectrodes of latest electrophysiological devices. We have presented an online spike sorting algorithm which extend an already well-established and highly used offline algorithm (SpyKING CIRCUS). We have shown that the spike trains of multiple neurons can be reconstructed in firm real-time. This new technology open the possibility to use adaptive sampling in closed-loop experiments to probe the spiking activity more efficiently. We presented a specific application on the retina where online spike sorting might be useful. We probed the selectivity of retinal ganglion cells to multiple perturbations in natural contexts. These results could be of great help for the design of neural prostheses such as artificial retinas. They could provide a better understanding of how the sensory neurons encode the information, which neural representation do they carry about the visual scene, and what are the visual features which are behaviorally relevant.

Summary in French

Introduction

Les neurosciences computationnelles cherchent à comprendre l'information que contiennent les signaux neuronaux. Des zones sensorielles aux zones motrices du système nerveux central, ces signaux se présentent sous une forme standard : une séquence d'impulsions électriques appelées potentiels d'action. Pour comprendre leur signification, nous devons comprendre la structure de ces séquences et élucider les règles qui définissent ce que l'on pourrait appeler la syntaxe neurale. Cette approche devrait révéler la manière dont nous recevons, traitons, stockons et transmettons les informations provenant du monde extérieur.

Au cours des dernières décennies, le progrès des techniques expérimentales ont ouvert de nouveaux horizons pour l'étude du code neural. Notre capacité à enregistrer l'activité d'une grande population de neurones s'est considérablement développée. C'est particulièrement vrai en électrophysiologie, où les implants en silicium et les réseaux de microélectrodes fournissent désormais un grand nombre et une forte densité de sites d'enregistrement. En conséquence, le volume de données neuronales enregistrées a énormément augmenté et remet en question les processus d'analyse de données utilisés habituellement.

Dans ce travail de thèse, je passe en revue les algorithmes de traitement du signal qui ont été proposés pour reconstruire indiciduellement l'activité électrique des neurones à partir des traces de potentiel extracellulaire enregistrées avec ces nouveaux dispositifs. Ce problème inverse de séparation de sources est appelé "tri de potentiel d'actions". J'identifie les propriétés communes et les principales différences de ces algorithms pour ensuite identifier les problèmes qui restent à résoudre. Dans la partie qui suit, je présente un outil générique pour le tri des potentiels d'actions en ligne, basée sur une combinaison de méthodes : le partitionnement par densité, la correspondance de motifs et le calcul parallèle. Cet outil permet de résoudre les trains de potentiels d'actions de chaque cellule enregistrée en temps réel.

La rétine est un système particulièrement adapté pour étudier comment les neurones sensoriels codent l'information visuelle. Ce système neuronal transforme les motifs lumineux qui entrent par l'oeil en séquences de potentiels d'actions qui sont ensuite envoyées au cerveau. L'information visuelle est transportée par ce système sensoriel avec quasiement aucune rétroaction de la part du cerveau, ce qui permet d'étudier le traitement visuel qu'il réalise de manière isolée. De plus, la planéité du tissu rétinien permet d'enregistrer facilement son activité à l'aide de réseaux de microélectrodes. Les réponses enregistrées à partir de grandes populations de cellules ganglionnaires, la sortie de la rétine, sont bruitées et présentent un degré de complexité élevé. En pratique, environ 30 types de cellules différents ont été identifiés et chacun d'entre eux est supposé coder une caractéristique différente de la scène visuelle. Ces représentations sont souvent non linéaires et s'adaptent aux statistiques visuelles telles que la luminance moyenne. Ces propriétés rendent le code rétinien difficile à déchiffrer et font l'objet d'une recherche active.

Dans une dernière partie, je présente une application spécifique sur la rétine où le tri en ligne des potentiels d'actions pourrait être utile. Classiquement, les cellules ganglionnaires sont supposées extraire des caractéristiques spécifiques de la scène visuelle telles que l'augmentation ou la diminution de la luminance, cellules ON ou OFF respectivement. Cependant, le traitement effectuée par la rétine dépend du contexte visuel. En utilisant une nouvelle approche perturbative, je montre qu'une même cellule peut être active ou inactive en fonction du contexte naturel. J'ai construit un modèle de réseau de neurones convolutionnel ajusté aux données qui peut récapituler cette dépendance au contexte. Les perturbations en ligne sont donc un outil prometteur pour étudier les calculs effectués au niveau de la rétine mais aussi plus largement dans d'autres systèmes sensoriels.

Cette thèse est organisée comme suit :

Chapitre 1 Je commence par une introduction de la physiologie de la rétine. Je montre que des fonctions complexes émergent de la diversité des types cellulaires et de la structure en réseau de ce système sensoriel.

Chapitre 2 Je passe en revue les algorithmes utilisés pour reconstruire individuellement l'activité des neurones à partir de traces de potentiel extracellulaire enregistrées avec des réseaux de microélectrodes denses et de grande échelle.

Chapitre 3 Je présente un ensemble générique d'outils pour le tri en ligne de potentiels d'action qui permet de reconstruire les trains de potentiel d'actions de chaque cellule enregistrée en temps réel.

Chapitre 4 J'étudie la stabilité de l'extraction de caractéristiques visuelles par les cellules ganglionnaires de la rétine dans différents contextes naturels avec une luminance et un contraste moyen fixes. En particulier, je montre que certaines cellules peuvent répondre de manière sélective à de petites augmentations de lumière dans certains contextes et à des diminutions dans d'autres.

Bibliography

Abadi, Martín, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng (2016). "Tensorflow: A system for large-scale machine learning". In: *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, pp. 265–283.

Aggarwal, Charu C, Jiawei Han, Jianyong Wang, and Philip S Yu (2003). "A Framework for Clustering Evolving Data Streams". In: *Proceedings of the 29th International Conference on Very Large Data Bases - Volume 29*. VLDB '03. Berlin, Germany: VLDB Endowment, pp. 81–92.

Amini, Amineh and Ty Wah (2011). "Density Micro-Clustering Algorithms on Data Streams: A Review". In: *Proceedings of the International Multi-Conference of Engineers and Computer Scientists* I, pp. 14–18.

Atick, Joseph J and A Norman Redlich (1992). "What does the retina know about natural scenes?" In: *Neural computation* 4.2, pp. 196–210.

Atiya, Amir F (1992). "Recognition of multiunit neural signals". In: *Biomedical Engineering, IEEE Transactions on* 39.7, pp. 723–729.

Attneave, Fred (1954). "Some informational aspects of visual perception." In: *Psychological review* 61.3, p. 183.

Baccus, Stephen A and Markus Meister (2002). "Fast and slow contrast adaptation in retinal circuitry". In: *Neuron* 36.5, pp. 909–919.

Baccus, Stephen A, Bence P Ölveczky, Mihai Manu, and Markus Meister (2008). "A retinal circuit that computes object motion". In: *Journal of Neuroscience* 28.27, pp. 6807–6817.

Baden, Tom, Philipp Berens, Katrin Franke, Miroslav Román Rosón, Matthias Bethge, and Thomas Euler (2016). "The functional diversity of retinal ganglion cells in the mouse". In: *Nature* 529.7586, pp. 345–350.

Baden, Tom, Thomas Euler, and Philipp Berens (2019). "Understanding the retinal basis of vision across species". In: *Nature Reviews Neuroscience*, pp. 1–16.

Bar-Gad, Izhar, Ya'acov Ritov, Eilon Vaadia, and Hagai Bergman (2001). "Failure in identification of overlapping spikes from multiple neuron activity causes artificial correlations". In: *Journal of neuroscience methods* 107.1, pp. 1–13.

Barlow, Horace B (1953). "Summation and inhibition in the frog's retina". In: *The Journal of physiology* 119.1, p. 69.

— (1961). "Possible principles underlying the transformation of sensory messages". In: *Sensory communication* 1, pp. 217–234.

Barlow, Horace B, Richard M Hill, and William R Levick (1964). "Retinal ganglion cells responding selectively to direction and speed of image motion in the rabbit". In: *The Journal of physiology* 173.3, p. 377.

Bashivan, Pouya, Kohitij Kar, and James J DiCarlo (2019). "Neural population control via deep image synthesis". In: *Science* 364.6439.

Ben-Ari, Mordechai (2006). *Principles of concurrent and distributed programming*. Pearson Education.

Benardete, Ethan A and Ehud Kaplan (1997). "The receptive field of the primate P retinal ganglion cell, I: Linear dynamics". In: *Visual neuroscience* 14.1, pp. 169–185.

Benda, Jan, Tim Gollisch, Christian K Machens, and Andreas VM Herz (2007). "From response to stimulus: adaptive sampling in sensory physiology". In: *Curr. Opin. Neurobiol.* 17.4, pp. 430–436.

Berdondini, Luca, PD Van Der Wal, Olivier Guenat, Nicolaas F de Rooij, Milena Koudelka-Hep, P Seitz, R Kaufmann, P Metzler, N Blanc, and S Rohr (2005). "High-density electrode array for imaging in vitro electrophysiological activity". In: *Biosensors and bioelectronics* 21.1, pp. 167–174.

Berry, Michael J and Markus Meister (1998). "Refractoriness and neural precision". In: *Advances in neural information processing systems*, pp. 110–116.

Bestel, Robert, Andreas W Daus, and Christiane Thielemann (2012). "A novel automated spike sorting algorithm with adaptable feature extraction". In: *Journal of neuroscience methods* 211.1, pp. 168–178.

Bölinger, Daniel and Tim Gollisch (2012). "Closed-Loop Measurements of Iso-Response Stimuli Reveal Dynamic Nonlinear Stimulus Integration in the Retina". In: *Neuron* 73.2, pp. 333–346.

Bomash, Illya, Yasser Roudi, and Sheila Nirenberg (2013). "A virtual retina for studying population coding". In: *PloS one* 8.1, e53363.

Borst, Alexander and Thomas Euler (2011). "Seeing things in motion: models, circuits, and mechanisms". In: *Neuron* 71.6, pp. 974–994.

Boycott, Brian Blundell and John E Dowling (1969). "Organization of the primate retina: light microscopy, with an appendix: a second type of midget bipolar cell in the primate retina". In: *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences* 255.799, pp. 109–184.

Brenner, Naama, William S Bialek, and Rob de Ruyter Van Steveninck (2000). "Adaptive rescaling maximizes information transmission". In: *Neuron* 26.3, pp. 695–702.

Brown, Glen D, Satoshi Yamada, and Terrence J Sejnowski (2001). "Independent component analysis at the neural cocktail party". In: *Trends in neurosciences* 24.1, pp. 54–63.

Brown, Solange P, Shigang He, and Richard H Masland (2000). "Receptive field microstructure and dendritic geometry of retinal ganglion cells". In: *Neuron* 27.2, pp. 371–383.

Buccino, Alessio Paolo, Cole Lincoln Hurwitz, Jeremy Magland, Samuel Garcia, Joshua H Siegle, Roger Hurwitz, and Matthias H Hennig (2019). "SpikeInterface, a unified framework for spike sorting". In: *bioRxiv*.

Cadena, Santiago A, George H Denfield, Edgar Y Walker, Leon A Gatys, Andreas S Tolias, Matthias Bethge, and Alexander S Ecker (2019). "Deep convolutional models improve predictions of macaque V1 responses to natural images". In: *PLoS computational biology* 15.4, e1006897.

Cai, Denise J, Daniel Aharoni, Tristan Shuman, Justin Shobe, Jeremy Biane, Weilin Song, Brandon Wei, Michael Veshkini, Mimi La-Vu, Jerry Lou, et al. (2016). "A shared neural ensemble links distinct contextual memories encoded close in time". In: *Nature* 534.7605, pp. 115–118.

Caldwell, John H, Nigel W Daw, and Harry J Wyatt (1978). "Effects of picrotoxin and strychnine on rabbit retinal ganglion cells: lateral interactions for cells with more complex receptive fields." In: *The Journal of physiology* 276.1, pp. 277–298.

Campagnola, Luke, Almar Klein, Eric Larson, Cyrille Rossant, and Nicolas P Rougier (2015). "VisPy: harnessing the GPU for fast, high-level visualization". In:

Cao, Feng, Martin Estert, Weining Qian, and Aoying Zhou (2006). "Density-based clustering over an evolving data stream with noise". In: *Proceedings of the 2006 SIAM international conference on data mining*. SIAM, pp. 328–339.

Chah, Ehsan, Vincent Hok, Andrea Della-Chiesa, John JH Miller, Shane M O'Mara, and Richard B Reilly (2011). "Automated spike sorting algo-

rithm based on Laplacian eigenmaps and k-means clustering". In: *Journal of neural engineering* 8.1, p. 016006.

Chamberland, Simon, Helen H Yang, Michael M Pan, Stephen W Evans, Sihui Guan, Mariya Chavarha, Ying Yang, Charleen Salesse, Haodi Wu, Joseph C Wu, et al. (2017). "Fast two-photon imaging of subcellular voltage dynamics in neuronal tissue with genetically encoded indicators". In: *Elife* 6, e25690.

Chichilnisky, Eduardo J (2001). "A simple white noise analysis of neuronal light responses". In: *Network: Computation in Neural Systems* 12.2, pp. 199–213.

Christie, Breanne P, Derek M Tat, Zachary T Irwin, Vikash Gilja, Paul Nuyujukian, Justin D Foster, Stephen I Ryu, Krishna V Shenoy, David E Thompson, and Cynthia a Chestek (2015). "Comparison of spike sorting and thresholding of voltage waveforms for intracortical brain-machine interface performance." In: *Journal of neural engineering* 12.1, p. 016009.

Cleland, Brian G, Mark W Dubin, and William R Levick (1971). "Sustained and transient neurones in the cat's retina and lateral geniculate nucleus". In: *The Journal of Physiology* 217.2, pp. 473–496.

Cowley, Benjamin, Ryan Williamson, Katerina Clemens, Matthew Smith, and M Yu Byron (2017). "Adaptive stimulus selection for optimizing neural population responses". In: *Advances in neural information processing systems*, pp. 1396–1406.

Curcio, Christine A and Anita E Hendrickson (1991). "Organization and development of the primate photoreceptor mosaic". In: *Progress in retinal research* 10, pp. 89–120.

Dacey, Dennis M (1993). "The mosaic of midget ganglion cells in the human retina". In: *Journal of Neuroscience* 13.12, pp. 5334–5355.

Daw, Nigel W (1968). "Colour-coded ganglion cells in the goldfish retina: extension of their receptive fields by means of new stimuli". In: *The Journal of physiology* 197.3, pp. 567–592.

De Monasterio, Francisco M (1978). "Properties of ganglion cells with atypical receptive-field organization in retina of macaques". In: *Journal of Neurophysiology* 41.6, pp. 1435–1449.

Demb, Jonathan B (2007). "Cellular mechanisms for direction selectivity in the retina". In: *Neuron* 55.2, pp. 179–186.

Demb, Jonathan B, Loren Haarsma, Michael A Freed, and Peter Sterling (1999). "Functional circuitry of the retinal ganglion cell's nonlinear receptive field". In: *Journal of Neuroscience* 19.22, pp. 9756–9767.

Deny, Stephane, Ulisse Ferrari, Emilie Mace, Pierre Yger, Romain Caplette, Serge Picaud, Gašper Tkačik, and Olivier Marre (2017). "Multiplexed

computations in retinal ganglion cells of a single type". In: *Nature communications* 8.1, pp. 1–17.

DiCarlo, James J, Davide Zoccolan, and Nicole C Rust (2012). "How does the brain solve visual object recognition?" In: *Neuron* 73.3, pp. 415–434.

Dugas, Charles, Yoshua Bengio, François Bélisle, Claude Nadeau, and René Garcia (2001). "Incorporating second-order functional knowledge for better option pricing". In: *Advances in neural information processing systems*, pp. 472–478.

Edin, Fredrik, Christian K Machens, Hartmut Schütze, and Andreas VM Herz (2004). "Searching for optimal sensory signals: iterative stimulus reconstruction in closed-loop experiments". In: *Journal of computational neuroscience* 17.1, pp. 47–56.

Egert, U, TH Knott, C Schwarz, M Nawrot, A Brandt, S Rotter, and M Diesmann (2002). "MEA-Tools: an open source toolbox for the analysis of multi-electrode data with MATLAB". In: *Journal of neuroscience methods* 117.1, pp. 33–42.

Einevoll, Gaute T, Felix Franke, Espen Hagen, Christophe Pouzat, and Kenneth D Harris (2012). "Towards reliable spike-train recordings from thousands of neurons with multielectrodes". In: *Current opinion in neurobiology* 22.1, pp. 11–17.

Ekanadham, Chaitanya, Daniel Tranchina, and Eero P Simoncelli (2011). "Recovery of sparse translation-invariant signals with continuous basis pursuit". In: *Signal Processing, IEEE Transactions on* 59.10, pp. 4735–4744.

— (2014). "A unified framework and method for automatic neural spike identification". In: *Journal of neuroscience methods* 222, pp. 47–55.

Enroth-Cugell, Christina and Alan W Freeman (1987). "The receptive-field spatial structure of cat retinal Y cells." In: *The Journal of Physiology* 384.1, pp. 49–79.

Enroth-Cugell, Christina and P Lennie (1975). "The control of retinal ganglion cell discharge by receptive field surrounds." In: *The Journal of Physiology* 247.3, pp. 551–578.

Enroth-Cugell, Christina and John G Robson (1966). "The contrast sensitivity of retinal ganglion cells of the cat". In: *The Journal of physiology* 187.3, pp. 517–552.

Ester, Martin, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu (1996). "A density-based algorithm for discovering clusters in large spatial databases with noise." In: *Kdd*. Vol. 96. 34, pp. 226–231.

Euler, Thomas, Peter B Detwiler, and Winfried Denk (2002). "Directionally selective calcium signals in dendrites of starburst amacrine cells". In: *Nature* 418.6900, pp. 845–852.

Farrow, Karl, Miguel Teixeira, Tamas Szikra, Tim J Viney, Kamill Balint, Keisuke Yonehara, and Botond Roska (2013). "Ambient illumination toggles a neuronal circuit switch in the retina and visual perception at cone threshold". In: *Neuron* 78.2, pp. 325–338.

Ferrari, Ulisse, Christophe Gardella, Olivier Marre, and Thierry Mora (2017). "Closed-Loop Estimation of Retinal Network Sensitivity by Local Empirical Linearization". In: *eNeuro* 4.6.

Field, David J (1987). "Relations between the statistics of natural images and the response properties of cortical cells". In: *Josa a* 4.12, pp. 2379–2394.

Fiscella, Michele, Karl Farrow, Ian L Jones, David Jäckel, Jan Müller, Urs Frey, Douglas J Bakkum, Péter Hantz, Botond Roska, and Andreas Hierlemann (2012). "Recording from defined populations of retinal ganglion cells using a high-density CMOS-integrated microelectrode array with real-time switchable electrode selection". In: *Journal of neuroscience methods* 211.1, pp. 103–113.

Fournier, Julien, Christian M Mueller, Mark Shein-Idelson, Mike Hemberger, and Gilles Laurent (2016). "Consensus-based sorting of neuronal spike waveforms". In: *PloS one* 11.8, e0160494.

Franke, Felix, Michal Natora, Clemens Boucsein, Matthias HJ Munk, and Klaus Obermayer (2010). "An online spike detection and spike classification algorithm capable of instantaneous resolution of overlapping spikes". In: *Journal of computational neuroscience* 29.1-2, pp. 127–148.

Franke, Felix, Robert Pröpper, Henrik Alle, Philipp Meier, Jörg RP Geiger, Klaus Obermayer, and Matthias HJ Munk (2015). "Spike sorting of synchronous spikes from local neuron ensembles". In: *Journal of neurophysiology* 114.4, pp. 2535–2549.

Franke, Felix, R Quian Quiroga, Andreas Hierlemann, and Klaus Obermayer (2015). "Bayes optimal template matching for spike sorting–combining fisher discriminant analysis with optimal filtering". In: *Journal of computational neuroscience* 38.3, pp. 439–459.

Fraser, George W, Steven M Chase, Andrew Whitford, and Andrew B Schwartz (2009). "Control of a brain–computer interface without spike sorting". In: *Journal of neural engineering* 6.5, p. 055004.

Freeman, Jeremy, Greg D Field, Peter H Li, Martin Greschner, Deborah E Gunning, Keith Mathieson, Alexander Sher, Alan M Litke, Liam Paninski, Eero P Simoncelli, et al. (2015). "Mapping nonlinear receptive field structure in primate retina at single cone resolution". In: *Elife* 4, e05241.

Friedman, Jerome H and John W Tukey (1974). "A projection pursuit algorithm for exploratory data analysis". In:

Gao, HongYing, Camille de Solages, and Clément Lena (2012). "Tetrode recordings in the cerebellar cortex". In: *Journal of Physiology-Paris* 106.3, pp. 128–136.

Garaschuk, Olga, Jennifer Linn, Jens Eilers, and Arthur Konnerth (2000). "Large-scale oscillatory calcium waves in the immature cortex". In: *Nature neuroscience* 3.5, pp. 452–459.

Geffen, Maria Neimark, Saskia EJ De Vries, and Markus Meister (2007). "Retinal ganglion cells can rapidly change polarity from Off to On". In: *PLoS Biol* 5.3, e65.

Gerstein, George L and William A Clark (1964). "Simultaneous studies of firing patterns in several neurons". In: *Science* 143.3612, pp. 1325–1327.

Gollisch, Tim and Andreas VM Herz (2012). "The iso-response method: measuring neuronal stimulus integration with closed-loop experiments". In: *Frontiers in neural circuits* 6, p. 104.

Gollisch, Tim and Markus Meister (2008a). "Modeling convergent ON and OFF pathways in the early visual system". In: *Biological cybernetics* 99.4-5, pp. 263–278.

— (2008b). "Rapid neural coding in the retina with relative spike latencies". In: *science* 319.5866, pp. 1108–1111.

— (2010). "Eye smarter than scientists believed: neural computations in circuits of the retina". In: *Neuron* 65.2, pp. 150–164.

Gozani, Shai N and John P Miller (1994). "Optimal discrimination and classification of neuronal action potential waveforms from multiunit, multichannel recordings using software-based linear filters". In: *IEEE Transactions on Biomedical Engineering* 41.4, pp. 358–372.

Gray, Charles M, Pedro E Maldonado, Mathew Wilson, and Bruce McNaughton (1995). "Tetrodes markedly improve the reliability and yield of multiple single-unit isolation from multi-unit recordings in cat striate cortex". In: *Journal of neuroscience methods* 63.1, pp. 43–54.

Grinvald, Amiram and Rina Hildesheim (2004). "VSDI: a new era in functional imaging of cortical dynamics". In: *Nature Reviews Neuroscience* 5.11, pp. 874–885.

Guenther, Frank H, Jonathan S Brumberg, E Joseph Wright, Alfonso Nieto-Castanon, Jason A Tourville, Mikhail Panko, Robert Law, Steven A Siebert, Jess L Bartels, Dinal S Andreasen, Princewill Ehirim, Hui Mao, and Philip R Kennedy (2009). "A wireless brain-machine interface for real-time speech synthesis". In: *PLoS ONE* 4.12.

Harris, Kenneth D, Darrell A Henze, Jozsef Csicsvari, Hajime Hirase, and György Buzsáki (2000). "Accuracy of tetrode spike separation as determined by simultaneous intracellular and extracellular measurements". In: *Journal of neurophysiology* 84.1, pp. 401–414.

Hartline, H Keffer (1938). "The response of single optic nerve fibers of the vertebrate eye to illumination of the retina". In: *American Journal of Physiology-Legacy Content* 121.2, pp. 400–415.

— (1940). "The receptive fields of optic nerve fibers". In: *American Journal of Physiology-Legacy Content* 130.4, pp. 690–699.

Hassenstein, Bernhard and Werner Reichardt (1956). "System-theoretical analysis of the time, sequence and sign evaluation in the movement perception of the r "u sselk ä fers chlorophanus". In: *magazine f "u r nature research B* 11.9-10, pp. 513–524.

Hattar, Samer, H-W Liao, Motoharu Takao, David M Berson, and K-W Yau (2002). "Melanopsin-containing retinal ganglion cells: architecture, projections, and intrinsic photosensitivity". In: *Science* 295.5557, pp. 1065–1070.

Heitman, Alexander, Nora Brackbill, Martin Greschner, Alexander Sher, Alan M Litke, and Eduardo J Chichilnisky (2016). "Testing pseudo-linear models of responses to natural scenes in primate retina". In: *bioRxiv*, p. 045336.

Hilgen, Gerrit, Martino Sorbaro, Sahar Pirmoradian, Jens-Oliver Muthmann, Ibolya Kepiro, Simona Ullo, Cesar Juarez Ramirez, Alessandro Maccione, Luca Berdondini, Vittorio Murino, Diego Sona, Francesca C Zanacchi, Evelyne Sernagor, and Matthias H Hennig (2016). "Unsupervised spike sorting for large scale, high density multielectrode arrays". In: *bioRxiv*, p. 048645.

Hintjens, Pieter (2010). *ZeroMQ: The Guide*. URL: `https://zguide.zeromq.org/`.

Hochstein, S and Robert M Shapley (1976). "Linear and nonlinear spatial subunits in Y cat retinal ganglion cells." In: *The Journal of physiology* 262.2, pp. 265–284.

Hosoya, Toshihiko, Stephen A Baccus, and Markus Meister (2005). "Dynamic predictive coding by the retina". In: *Nature* 436.7047, pp. 71–77.

Hsu, Anne, Alexander Borst, and Frédéric E Theunissen (2004). "Quantifying variability in neural responses and its application for the validation of model predictions". In: *Network: Computation in Neural Systems* 15.2, pp. 91–109.

Hubel, David H (1957). "Tungsten microelectrode for recording from single units". In: *Science* 125.3247, pp. 549–550.

Hulata, Eyal, Ronen Segev, and Eshel Ben-Jacob (2002). "A method for spike sorting and detection based on wavelet packets and Shannon's mutual information". In: *Journal of neuroscience methods* 117.1, pp. 1–12.

Jäckel, David, Urs Frey, Michele Fiscella, Felix Franke, and Andreas Hierlemann (2012). "Applicability of independent component analysis on high-

density microelectrode array recordings". In: *Journal of neurophysiology* 108.1, pp. 334–348.

Johnston, Jamie, Huayu Ding, Sofie H Seibel, Federico Esposti, and Leon Lagnado (2014). "Rapid mapping of visual receptive fields by filtered back projection: application to multi-neuronal electrophysiology and imaging". In: *The Journal of Physiology* 592.22, pp. 4839–4854.

Jones, Kelly E, Patrick K Campbell, and Richard A Normann (1992). "A glass/silicon composite intracortical electrode array". In: *Annals of biomedical engineering* 20.4, pp. 423–437.

Jun, James J., Nicholas A. Steinmetz, Joshua H. Siegle, Daniel J. Denman, Marius Bauza, Brian Barbarits, Albert K. Lee, Costas A. Anastassiou, Alexandru Andrei, Çağatay Aydin, Mladen Barbic, Timothy J. Blanche, Vincent Bonin, João Couto, Barundeb Dutta, Sergey L. Gratiy, Diego A. Gutnisky, Michael Häusser, Bill Karsh, Peter Ledochowitsch, Carolina Mora Lopez, Catalin Mitelut, Silke Musa, Michael Okun, Marius Pachitariu, Jan Putzeys, P. Dylan Rich, Cyrille Rossant, Wei Lung Sun, Karel Svoboda, Matteo Carandini, Kenneth D. Harris, Christof Koch, John O'Keefe, and Timothy D. Harris (2017). "Fully integrated silicon probes for high-density recording of neural activity". In: *Nature*.

Jun, James J, Catalin Mitelut, Chongxi Lai, Sergey L Gratiy, Costas A Anastassiou, and Timothy D Harris (2017). "Real-time spike sorting platform for high-density extracellular probes with ground-truth validation and drift correction". In: pp. 1–29.

Kadir, Shabnam N, Dan FM Goodman, and Kenneth D Harris (2014). "High-dimensional cluster analysis with the masked EM algorithm". In: *Neural computation*.

Kandel, Eric R, James H Schwartz, Thomas M Jessell, Department of Biochemistry, Molecular Biophysics Thomas Jessell, Steven Siegelbaum, and AJ Hudspeth (2000). *Principles of neural science*. Vol. 4. McGraw-hill New York.

Kastner, David B, Stephen A Baccus, and Tatyana O Sharpee (2015). "Critical and maximally informative encoding between neural populations in the retina". In: *Proceedings of the National Academy of Sciences* 112.8, pp. 2533–2538.

Keshishian, Menoua, Hassan Akbari, Bahar Khalighinejad, Jose L Herrero, Ashesh D Mehta, and Nima Mesgarani (2020). "Estimating and interpreting nonlinear receptive field of sensory neural responses with deep neural network models". In: *eLife* 9, e53445.

Kim, In-Jung, Yifeng Zhang, Masahito Yamagata, Markus Meister, and Joshua R Sanes (2008). "Molecular identification of a retinal cell type that responds to upward motion". In: *Nature* 452.7186, pp. 478–482.

Kim, Kyung Hwan and Sung June Kim (2000). "Neural spike sorting under nearly 0-dB signal-to-noise ratio using nonlinear energy operator and artificial neural-network classifier". In: *Biomedical Engineering, IEEE Transactions on* 47.10, pp. 1406–1411.

Klindt, David, Alexander S Ecker, Thomas Euler, and Matthias Bethge (2017). "Neural system identification for large populations separating "what" and "where"". In: *Advances in Neural Information Processing Systems*, pp. 3506–3516.

Korenberg, Michael J, Hiroko M Sakai, and Ken-ichi Naka (1989). "Dissection of the neuron network in the catfish inner retina. III. Interpretation of spike kernels". In: *Journal of Neurophysiology* 61.6, pp. 1110–1120.

Kuffler, Stephen W (1953). "Discharge patterns and functional organization of mammalian retina". In: *Journal of neurophysiology* 16.1, pp. 37–68.

Lambacher, Armin, Martin Jenkner, Matthias Merz, Björn Eversmann, R Alexander Kaul, Franz Hofmann, Roland Thewes, and Peter Fromherz (2004). "Electrical imaging of neuronal activity by multi-transistor-array (MTA) recording at 7.8 μm resolution". In: *Applied Physics A* 79.7, pp. 1607–1611.

LeCun, Yann, Léon Bottou, Yoshua Bengio, and Patrick Haffner (1998). "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11, pp. 2278–2324.

Lee, JinHyung, Catalin Mitelut, Hooshmand Shokri, Ian Kinsella, Nishchal Dethe, Shenghao Wu, Kevin Li, Eduardo Blancas Reyes, Denis Turcu, Eleanor Batty, Young Joon Kim, Nora Brackbill, Alexandra Kling, Georges Goetz, Eduardo J Chichilnisky, David Carlson, and Liam Paninski (2020). "YASS: Yet Another Spike Sorter applied to large-scale multi-electrode array recordings in primate retina". In: *bioRxiv*.

Lefebvre, Baptiste, Pierre Yger, and Olivier Marre (2016). "Recent progress in multi-electrode spike sorting methods". In: *Journal of Physiology Paris* 110.4, pp. 327–335.

Leibig, Christian, Thomas Wachtler, and Günther Zeck (2016). "Unsupervised neural spike sorting for high-density microelectrode arrays with convolutive independent component analysis". In: *Journal of Neuroscience Methods* 271, pp. 1–13.

Letelier, Juan Carlos and Pamela P Weber (2000). "Spike sorting based on discrete wavelet transform coefficients". In: *Journal of neuroscience methods* 101.2, pp. 93–106.

Lettvin, Jerome Y, Humberto R Maturana, Warren S McCulloch, and Walter H Pitts (1959). "What the frog's eye tells the frog's brain". In: *Proceedings of the IRE* 47.11, pp. 1940–1951.

Levick, William R (1967). "Receptive fields and trigger features of ganglion cells in the visual streak of the rabbit's retina". In: *The Journal of physiology* 188.3, p. 285.

Lewi, Jeremy, Robert Butera, and Liam Paninski (2007). "Real-time adaptive information-theoretic optimization of neurophysiology experiments". In: *Advances in Neural Information Processing Systems*, pp. 857–864.

— (2009). "Sequential optimal design of neurophysiology experiments". In: *Neural computation* 21.3, pp. 619–687.

Lewicki, Michael S (1998). "A review of methods for spike sorting: the detection and classification of neural action potentials". In: *Network: Computation in Neural Systems* 9.4, R53–R78.

Lin, Michael Z and Mark J Schnitzer (2016). "Genetically encoded indicators of neuronal activity". In: *Nature neuroscience* 19.9, p. 1142.

Litke, Alan M, Nathan Bezayiff, Eduardo J Chichilnisky, W Cunningham, Wladyslaw Dabrowski, Alexander A Grillo, Matthew V Grivich, Pawel Grybos, Pawel Hottowy, Sergei Kachiguine, Rachel S Kalmar, Keith Mathieson, Dumitru Petrusca, M Rahman, and Alexander Sher (2004). "What does the eye tell the brain?: Development of a system for the large-scale recording of retinal output activity". In: *IEEE Transactions on Nuclear Science* 51.4, pp. 1434–1440.

Logothetis, Nikos K and David L Sheinberg (1996). "Visual object recognition". In: *Annual review of neuroscience* 19.1, pp. 577–621.

Machens, Christian K (2002). "Adaptive sampling by information maximization". In: *Physical Review Letters* 88.22, p. 228104.

Machens, Christian K, Tim Gollisch, Olga Kolesnikova, and Andreas VM Herz (2005). "Testing the efficiency of sensory coding with optimal stimulus ensembles". In: *Neuron* 47.3, pp. 447–456.

MacNeil, Margaret A and Richard H Masland (1998). "Extreme diversity among amacrine cells: implications for function". In: *Neuron* 20.5, pp. 971–982.

Magland, Jeremy, James J Jun, Elizabeth Lovero, Alexander J Morley, Cole Lincoln Hurwitz, Alessio Paolo Buccino, Samuel Garcia, and Alex H Barnett (2020). "Spikeforest, reproducible web-facing ground-truth validation of automated neural spike sorters". In: *eLife*.

Maheswaranathan, Niru, Lane T McIntosh, David B Kastner, Josh Melander, Luke Brezovec, Aran Nayebi, Julia Wang, Surya Ganguli, and Stephen A Baccus (2018). "Deep learning models reveal internal structure and diverse computations in the retina under natural scenes". In: *bioRxiv*, p. 340943.

Maheswaranathan, Niru, Lane T McIntosh, Hidenori Tanaka, Satchel Grant, David B. Kastner, Josh B. Melander, Aran Nayebi, Luke Brezovec, Ju-

lia Wang, Surya Ganguli, and Stephen A. Baccus (2019). "The dynamic neural code of the retina for natural scenes". In: *bioRxiv*, p. 340943.

Mallat, Stéphane G and Zhifeng Zhang (1993). "Matching pursuits with time-frequency dictionaries". In: *Signal Processing, IEEE Transactions on* 41.12, pp. 3397–3415.

Markram, Henry, Joachim Lübke, Michael Frotscher, and Bert Sakmann (1997). "Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs". In: *Science* 275.5297, pp. 213–215.

Marre, Olivier, Dario Amodei, Nikhil Deshmukh, Kolia Sadeghi, Frederick Soo, Timothy E Holy, and Michael J Berry (2012). "Mapping a complete neural population in the retina". In: *The Journal of Neuroscience* 32.43, pp. 14859–14873.

Masland, Richard H (2012). "The neuronal organization of the retina". In: *Neuron* 76.2, pp. 266–280.

McGill, Kevin C and Leslie J Dorfman (1984). "High-resolution alignment of sampled waveforms". In: *IEEE transactions on biomedical engineering* 6, pp. 462–468.

McIntosh, Lane T, Niru Maheswaranathan, Aran Nayebi, Surya Ganguli, and Stephen Baccus (2016). "Deep learning models of the retinal response to natural scenes". In: *Advances in neural information processing systems*, pp. 1369–1377.

McNaughton, Bruce L, John O'Keefe, and Carol A Barnes (1983). "The stereotrode: a new technique for simultaneous isolation of several single units in the central nervous system from multiple unit records". In: *Journal of neuroscience methods* 8.4, pp. 391–397.

Meister, Markus and Michael J Berry (1999). "The neural code of the retina". In: *Neuron* 22.3, pp. 435–450.

Meister, Markus, Jerome Pine, and Denis A Baylor (1994). "Multi-neuronal signals from the retina: acquisition and analysis". In: *Journal of neuroscience methods* 51.1, pp. 95–106.

Mills, Stephen L and Stephen C Massey (1995). "Differential properties of two gap junctional pathways made by AII amacrine cells". In: *Nature* 377.6551, pp. 734–737.

Müller, Jan, Marco Ballini, Paolo Livi, Yihui Chen, Milos Radivojevic, Amir Shadmani, Vijay Viswam, Ian L Jones, Michele Fiscella, Roland Diggelmann, Alexander Stettler, Urs Frey, Douglas J Bakkum, and Hierlemann Andreas (2015). "High-resolution CMOS MEA platform to study neurons at subcellular, cellular, and network levels". In: *Lab on a Chip* 15.13, pp. 2767–2780.

Münch, Thomas A, Rava Azeredo Da Silveira, Sandra Siegert, Tim James Viney, Gautam B Awatramani, and Botond Roska (2009). "Approach

sensitivity in the retina processed by a multifunctional neural circuit". In: *Nature neuroscience* 12.10, pp. 1308–1316.

Nath, Amurta and Gregory W Schwartz (2016). "Cardinal orientation selectivity is represented by two distinct ganglion cell types in mouse retina". In: *Journal of Neuroscience* 36.11, pp. 3208–3221.

Navajas, Joaquin, Deren Y Barsakcioglu, Amir Eftekhar, Andrew Jackson, and G Timothy (2014). "Minimum Requirements for Accurate and Efficient Real-Time On- Chip Spike Sorting". In: *Journal of neuroscience methods* 230, pp. 51–64.

Neto, Joana P, Gonçalo Lopes, João Frazão, Joana Nogueira, Pedro Lacerda, Pedro Baião, Arno Aarts, Alexandru Andrei, Silke Musa, Elvira Fortunato, Pedro Barquinha, and Adam R Kampff (2016). "Validating silicon polytrodes with paired juxtacellular recordings: method and dataset". In: *bioRxiv*, p. 037937.

Nguyen, T K T, Z Navratilova, H Cabral, L Wang, G Gielen, F P Battaglia, and C Bartic (2014). "Closed-loop optical neural stimulation based on a 32-channel low-noise recording system with online spike sorting." In: *Journal of neural engineering* 11.4, p. 046005.

Nicolelis, Miguel AL, Dragan Dimitrov, Jose M Carmena, Roy Crist, Gary Lehew, Jerald D Kralik, and Steven P Wise (2003). "Chronic, multisite, multielectrode recordings in macaque monkeys". In: *Proceedings of the National Academy of Sciences* 100.19, pp. 11041–11046.

Nirenberg, Sheila, Steve M Carcieri, Adam L Jacobs, and Peter E Latham (2001). "Retinal ganglion cells act largely as independent encoders". In: *Nature* 411.6838, pp. 698–701.

Oliynyk, Andriy, Claudio Bonifazzi, Fernando Montani, and Luciano Fadiga (2012). "Automatic online spike sorting with singular value decomposition and fuzzy C-mean clustering". In: *BMC neuroscience* 13.1, p. 1.

Ölveczky, Bence P, Stephen A Baccus, and Markus Meister (2003). "Segregation of object and background motion in the retina". In: *Nature* 423.6938, pp. 401–408.

— (2007). "Retinal adaptation to object motion". In: *Neuron* 56.4, pp. 689–700.

Oyster, Clyde W and Horace B Barlow (1967). "Direction-selective units in rabbit retina: distribution of preferred directions". In: *Science* 155.3764, pp. 841–842.

Pachitariu, Marius, Nicholas Steinmetz, Shabnam Kadir, Matteo Carandini, and Kenneth D Harris (2016). "Kilosort: realtime spike-sorting for extracellular electrophysiology with hundreds of channels". In: *bioRxiv*, p. 061481.

Paninski, Liam (2004). "Maximum likelihood estimation of cascade point-process neural encoding models". In: *Network: Computation in Neural Systems* 15.4, pp. 243–262.

— (2005). "Asymptotic theory of information-theoretic experimental design". In: *Neural Computation* 17.7, pp. 1480–1507.

Park, Jongkil, Gookhwa Kim, and Sang-Don Jung (2017). "A 128-channel FPGA-based real-time spike-sorting bidirectional closed-loop neural interface system". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 25.12, pp. 2227–2238.

Park, Mijung and Jonathan W Pillow (2012). "Bayesian active learning with localized priors for fast receptive field characterization". In: *Advances in neural information processing systems*, pp. 2348–2356.

Passaglia, Christopher L, John B Troy, Lukas Rüttiger, and Barry B Lee (2002). "Orientation sensitivity of ganglion cells in primate retina". In: *Vision research* 42.6, pp. 683–694.

Paszke, Adam, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer (2017). "Automatic differentiation in pytorch". In:

Paszke, Adam, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach De Vito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala (2019). "Pytorch: An imperative style, high-performance deep learning library". In: *Advances in neural information processing systems*, pp. 8026–8037.

Pavlov, Alexey, Valeri A Makarov, Ioulia Makarova, and Fivos Panetsos (2007). "Sorting of neural spikes: when wavelet based methods outperform principal component analysis". In: *Natural Computing* 6.3, pp. 269–281.

Pearson, James T and Daniel Kerschensteiner (2015). "Ambient illumination switches contrast preference of specific retinal processing streams". In: *Journal of neurophysiology* 114.1, pp. 540–550.

Pedreira, Carlos, Juan Martinez, Matias J Ison, and R Quian Quiroga (2012). "How many neurons can we see with current spike sorting algorithms?" In: *Journal of neuroscience methods* 211.1, pp. 58–65.

Peichl, Leo and Heinz Wässle (1979). "Size, scatter and coverage of ganglion cell receptive field centres in the cat retina." In: *The Journal of physiology* 291.1, pp. 117–141.

Petrusca, Dumitru, Matthew I Grivich, Alexander Sher, Greg D Field, Jeffrey L Gauthier, Martin Greschner, Jonathon Shlens, Eduardo J Chichilnisky, and Alan M Litke (2007). "Identification and characterization of a Y-like

primate retinal ganglion cell type". In: *Journal of Neuroscience* 27.41, pp. 11019–11027.

Pillow, Jonathan W, Jonathon Shlens, Eduardo J Chichilnisky, and Eero P Simoncelli (2013). "A model-based spike sorting algorithm for removing correlation artifacts in multi-neuron recordings". In: *PloS one* 8.5, e62123.

Pillow, Jonathan W, Jonathon Shlens, Liam Paninski, Alexander Sher, Alan M Litke, Eduardo J Chichilnisky, and Eero P Simoncelli (2008). "Spatio-temporal correlations and visual signalling in a complete neuronal population". In: *Nature* 454.7207, pp. 995–999.

Pitkow, Xaq and Markus Meister (2012). "Decorrelation and efficient coding by retinal ganglion cells". In: *Nature neuroscience* 15.4, pp. 628–635.

Ponce, Carlos R, Will Xiao, Peter F Schade, Till S Hartmann, Gabriel Kreiman, and Margaret S Livingstone (2019). "Evolving images for visual neurons using a deep generative network reveals coding principles and neuronal preferences". In: *Cell* 177.4, pp. 999–1009.

Pouzat, Christophe, Ofer Mazor, and Gilles Laurent (2002). "Using noise signature to optimize spike-sorting and to assess neuronal classification quality". In: *Journal of neuroscience methods* 122.1, pp. 43–57.

Prentice, Jason S, Jan Homann, Kristina D Simmons, Gašper Tkačik, Vijay Balasubramanian, and Philip C Nelson (2011). "Fast, scalable, bayesian spike identification for multi-electrode arrays". In: *PloS one* 6.7, e19884.

Prochazka, VJ, Bastian Conrad, and Franz Sindermann (1972). "A neuroelectric signal recognition system". In: *Electroencephalography and clinical neurophysiology* 32.1, pp. 95–97.

Provencio, Ignacio, Ignacio R Rodriguez, Guisen Jiang, William Pär Hayes, Ernesto F Moreira, and Mark D Rollag (2000). "A novel human opsin in the inner retina". In: *Journal of Neuroscience* 20.2, pp. 600–605.

Quiroga, R Quian, Zoltan Nadasdy, and Yoram Ben-Shaul (2004). "Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering". In: *Neural computation* 16.8, pp. 1661–1687.

Quiroga, R Quian, Leila Reddy, Christof Koch, and Itzhak Fried (2007). "Decoding visual inputs from multiple neurons in the human temporal lobe". In: *Journal of neurophysiology* 98.4, pp. 1997–2007.

Rey, Hernan Gonzalo, Carlos Pedreira, and R Quian Quiroga (2015). "Past, present and future of spike sorting techniques". In: *Brain research bulletin* 119, pp. 106–117.

Rieke, Foster and Denis A Baylor (1998). "Single-photon detection by rod cells of the retina". In: *Reviews of Modern Physics* 70.3, p. 1027.

Rieke, Fred and Michael E Rudd (2009). "The challenges natural images pose for visual adaptation". In: *Neuron* 64.5, pp. 605–616.

Rieke, Fred, David Warland, Rob De Ruyter Van Steveninck, and William S Bialek (1999). *Spikes: exploring the neural code*. Vol. 7. 1. MIT press Cambridge.

Roberts, William M and Daniel K Hartline (1975). "Separation of multi-unit nerve impulse trains by a multi-channel linear filter algorithm". In: *Brain research* 94.1, pp. 141–149.

Rodieck, Robert W (1965). "Quantitative analysis of cat retinal ganglion cell response to visual stimuli". In: *Vision research* 5.12, pp. 583–601.

— (1967). "Receptive fields in the cat retina: a new type". In: *Science* 157.3784, pp. 90–92.

Rodieck, Robert W and Jonathan Stone (1965a). "Analysis of receptive fields of cat retinal ganglion cells". In: *Journal of neurophysiology* 28.5, pp. 833–849.

— (1965b). "Response of cat retinal ganglion cells to moving visual patterns". In: *Journal of Neurophysiology* 28.5, pp. 819–832.

Roska, Botond and Frank Werblin (2001). "Vertical interactions across ten parallel, stacked representations in the mammalian retina". In: *Nature* 410.6828, pp. 583–587.

Rossant, Cyrille and Kenneth D Harris (2013). "Hardware-accelerated interactive data visualization for neuroscience in Python". In: *Frontiers in neuroinformatics* 7.

Rossant, Cyrille, Shabnam N Kadir, Dan FM Goodman, John Schulman, Maximilian LD Hunter, Aman B Saleem, Andres Grosmark, Mariano Belluscio, George H Denfield, Alexander S Ecker, Andreas S Tolias, Samuel Solomon, György Buzsàki, Matteo Carandini, and Kenneth D Harris (2016). "Spike sorting for large, dense electrode arrays". In: *Nature neuroscience* 19, pp. 634–641.

Rowe, Michael H and Jonathan Stone (1980). "Parametric and feature extraction analyses of the receptive fields of visual neurones". In: *Brain, behavior and evolution* 17.2, pp. 103–122.

Ruderman, Daniel L and William S Bialek (1994). "Statistics of natural images: Scaling in the woods". In: *Advances in neural information processing systems*, pp. 551–558.

Rutishauser, Ueli, Erin M Schuman, and Adam N Mamelak (2006). "Online detection and sorting of extracellularly recorded action potentials in human medial temporal lobe recordings, in vivo". In: *Journal of Neuroscience Methods* 154.1-2, pp. 204–224.

Saeed, Maryam and Awais M Kamboh (2013). "Hardware architecture for on-chip unsupervised online neural spike sorting". In: *International IEEE/EMBS Conference on Neural Engineering, NER*.

Saito, Hide-Aki (1983). "Morphology of physiologically identified X-, Y-, and W-type retinal ganglion cells of the cat". In: *Journal of Comparative Neurology* 221.3, pp. 279–288.

Sakai, Hiroko M and Ken-ichi Naka (1992). "Response dynamics and receptive-field organization of catfish amacrine cells". In: *Journal of neurophysiology* 67.2, pp. 430–442.

Sakai, Hiroko M, Ken-ichi Naka, and Michael J Korenberg (1988). "White-noise analysis in visual neuroscience". In: *Visual neuroscience* 1.3, pp. 287–296.

Sakai, Hiroko M, JL Wang, and Ken-ichi Naka (1995). "Contrast gain control in the lower vertebrate retinas." In: *The Journal of general physiology* 105.6, pp. 815–835.

Sakmann, Bert and Otto D Creutzfeldt (1969). "Scotopic and mesopic light adaptation in the cat's retina". In: *Pflügers Archiv* 313.2, pp. 168–185.

Schoppe, Oliver, Nicol S Harper, Ben DB Willmore, Andrew J King, and Jan WH Schnupp (2016). "Measuring the performance of neural models". In: *Frontiers in computational neuroscience* 10, p. 10.

Schwartz, Gregory W, Haruhisa Okawa, Felice A Dunn, Josh L Morgan, Daniel Kerschensteiner, Rachel O Wong, and Fred Rieke (2012). "The spatial structure of a nonlinear receptive field". In: *Nature neuroscience* 15.11, pp. 1572–1580.

Schwartz, Odelia, Jonathan W Pillow, Nicole C Rust, and Eero P Simoncelli (2006). "Spike-triggered neural characterization". In: *Journal of vision* 6.4, pp. 13–13.

Segev, Ronen, Joe Goodhouse, Jason Puchalla, and Michael J Berry (2004). "Recording spikes from a large fraction of the ganglion cells in a retinal patch". In: *Nature neuroscience* 7.10, pp. 1155–1162.

Shan, Kevin Q, Evgueniy V Lubenov, and Athanassios G Siapas (2017). "Model-based spike sorting with a mixture of drifting t -distributions". In: *bioRxiv*.

Sharpee, Tatyana, Nicole C Rust, and William S Bialek (2004). "Analyzing neural responses to natural signals: maximally informative dimensions". In: *Neural computation* 16.2, pp. 223–250.

Shoham, Shy, Matthew R Fellows, and Richard A Normann (2003). "Robust, automatic spike sorting using mixtures of multivariate t-distributions". In: *Journal of neuroscience methods* 127.2, pp. 111–122.

Shreiner, Dave, Graham Sellers, John Kessenich, and Bill Licea-Kane (2013). *OpenGL programming guide: The Official guide to learning OpenGL, version 4.3*. Addison-Wesley.

Silva, Jonathan A, Elaine R Faria, Rodrigo C Barros, Eduardo R Hruschka, André CPLF de Carvalho, and João Gama (July 2013). "Data Stream Clustering: A Survey". In: *ACM Comput. Surv.* 46.1.

Silveira, Rava Azeredo da and Botond Roska (2011). "Cell types, circuits, computation". In: *Current opinion in neurobiology* 21.5, pp. 664–671.

Simon, William (1965). "The real-time sorting of neuro-electric action potentials in multiple unit studies". In: *Electroencephalography and clinical neurophysiology* 18.2, pp. 192–195.

Simoncelli, Eero P and Bruno A Olshausen (2001). "Natural image statistics and neural representation". In: *Annual review of neuroscience* 24.1, pp. 1193–1216.

Sitaram, Ranganatha, Tomas Ros, Luke Stoeckel, Sven Haller, Frank Scharnowski, Jarrod Lewis-Peacock, Nikolaus Weiskopf, Maria Laura Blefari, Mohit Rana, Ethan Oblak, Niels Birbaumer, and James Sulzer (2017). "Closed-loop brain training: the science of neurofeedback". In: *Nature Reviews Neuroscience* 18.2, pp. 86–100.

Smirnakis, Stelios M, Michael J Berry, David K Warland, William S Bialek, and Markus Meister (1997). "Adaptation of retinal processing to image contrast and spatial scale". In: *Nature* 386.6620, pp. 69–73.

Soo, Frederick S, Gregory W Schwartz, Kolia Sadeghi, and Michael J Berry (2011). "Fine spatial information represented in a population of retinal ganglion cells". In: *Journal of Neuroscience* 31.6, pp. 2145–2155.

Spampinato, Giulia LB, Elric Esposito, Pierre Yger, Jens Duebel, Serge Picaud, and Olivier Marre (Mar. 2018). *Ground truth recordings for validation of spike sorting algorithms*. Version 1.0.

Srinivasan, Mandyam Veerambudi, Simon Barry Laughlin, and Andreas Dubs (1982). "Predictive coding: a fresh view of inhibition in the retina". In: *Proceedings of the Royal Society of London. Series B. Biological Sciences* 216.1205, pp. 427–459.

Stosiek, Christoph, Olga Garaschuk, Knut Holthoff, and Arthur Konnerth (2003). "In vivo two-photon calcium imaging of neuronal networks". In: *Proceedings of the National Academy of Sciences* 100.12, pp. 7319–7324.

Sun, Wenzhi, Qiudong Deng, William R Levick, and Shigang He (2006). "ON direction-selective ganglion cells in the mouse retina". In: *The Journal of physiology* 576.1, pp. 197–202.

Swindale, Nicholas V and Martin A Spacek (2014). "Spike sorting for polytrodes: a divide and conquer approach". In: *Frontiers in systems neuroscience* 8.

— (2015). "Spike detection methods for polytrodes and high density microelectrode arrays". In: *Journal of computational neuroscience* 38.2, pp. 249–261.

Takahashi, Susumu, Yuichiro Anzai, and Yoshio Sakurai (2003). "Automatic sorting for multi-neuronal activity recorded with tetrodes in the presence of overlapping spikes". In: *Journal of neurophysiology* 89.4, pp. 2245–2258.

Taylor, W Rowland and David I Vaney (2003). "New directions in retinal research". In: *Trends in neurosciences* 26.7, pp. 379–385.

Theis, Lucas, Philipp Berens, Emmanouil Froudarakis, Jacob Reimer, Miroslav Román Rosón, Tom Baden, Thomas Euler, Andreas S Tolias, and Matthias Bethge (2016). "Benchmarking spike rate inference in population calcium imaging". In: *Neuron* 90.3, pp. 471–482.

Thorpe, Simon, Denis Fize, and Catherine Marlot (1996). "Speed of processing in the human visual system". In: *nature* 381.6582, pp. 520–522.

Tien, Nai-Wen, James T Pearson, Charles R Heller, Jay Demas, and Daniel Kerschensteiner (2015). "Genetically identified suppressed-by-contrast retinal ganglion cells reliably signal self-generated visual stimuli". In: *Journal of Neuroscience* 35.30, pp. 10815–10820.

Tikidji-Hamburyan, Alexandra, Katja Reinhard, Hartwig Seitter, Anahit Hovhannisyan, Christopher A Procyk, Annette E Allen, Martin Schenk, Robert J Lucas, and Thomas A Münch (2015). "Retinal output changes qualitatively with every change in ambient illuminance". In: *Nature neuroscience* 18.1, pp. 66–74.

Todorova, Sonia, Patrick Sadtler, Aaron Batista, Steven Chase, and Valérie Ventura (2014). "To sort or not to sort: the impact of spike-sorting on neural decoding performance". In: *Journal of neural engineering* 11.5, p. 056005.

Trautmann, Eric M, Sergey D Stavisky, Subhaneil Lahiri, Katherine C Ames, Matthew T Kaufman, Daniel J O'Shea, Saurabh Vyas, Xulu Sun, Stephen I Ryu, Surya Ganguli, et al. (2019). "Accurate estimation of neural population dynamics without spike sorting". In: *Neuron* 103.2, pp. 292–308.

Tu, Daniel C, Dongyang Zhang, Jay Demas, Elon B Slutsky, Ignacio Provencio, Timothy E Holy, and Russell N Van Gelder (2005). "Physiologic diversity and development of intrinsically photosensitive retinal ganglion cells". In: *Neuron* 48.6, pp. 987–999.

Turner, Maxwell H and Fred Rieke (2016). "Synaptic rectification controls nonlinear spatial integration of natural visual inputs". In: *Neuron* 90.6, pp. 1257–1271.

Valencia, Daniel and Amirhossein Alimohammad (2019). "An Efficient Hardware Architecture for Template Matching-Based Spike Sorting". In: *IEEE Transactions on Biomedical Circuits and Systems*.

Van Hateren, J Hans and Arjen van der Schaaf (1998). "Independent component filters of natural images compared with simple cells in primary

visual cortex". In: *Proceedings of the Royal Society of London. Series B: Biological Sciences* 265.1394, pp. 359–366.

Van Wyk, Michiel, Heinz Wässle, and W Rowland Taylor (2009). "Receptive field properties of ON-and OFF-ganglion cells in the mouse retina". In: *Visual neuroscience* 26.3, p. 297.

Vaney, David I, Benjamin Sivyer, and W Rowland Taylor (2012). "Direction selectivity in the retina: symmetry and asymmetry in structure and function". In: *Nature Reviews Neuroscience* 13.3, pp. 194–208.

Victor, Jonathan D (1987). "The dynamics of the cat retinal X cell centre." In: *The Journal of physiology* 386.1, pp. 219–246.

Victor, Jonathan D and Robert M Shapley (1979). "The nonlinear pathway of Y ganglion cells in the cat retina." In: *The Journal of General Physiology* 74.6, pp. 671–689.

Walker, Edgar Y, Fabian H Sinz, Erick Cobos, Taliah Muhammad, Emmanouil Froudarakis, Paul G Fahey, Alexander S Ecker, Jacob Reimer, Xaq Pitkow, and Andreas S Tolias (2019). "Inception loops discover what excites neurons most using deep predictive models". In: *Nature neuroscience* 22.12, pp. 2060–2065.

Wang, Shengjie, Abdel-rahman Mohamed, Rich Caruana, Jeff Bilmes, Matthai Plilipose, Matthew Richardson, Krzysztof Geras, Gregor Urban, and Ozlem Aslan (2016). "Analysis of deep neural networks with extended data jacobian matrix". In: *International Conference on Machine Learning*, pp. 718–726.

Wässle, Heinz (2004). "Parallel processing in the mammalian retina". In: *Nature Reviews Neuroscience* 5.10, pp. 747–757.

Watson, Andrew B (2014). "A formula for human retinal ganglion cell receptive field density as a function of visual field location". In: *Journal of vision* 14.7, pp. 15–15.

Wiesel, Torsten N (1960). "Receptive fields of ganglion cells in the cat's retina". In: *The Journal of Physiology* 153.3, p. 583.

Wood, E, M Fellows, JR Donoghue, and MJ Black (2004). "Automatic spike sorting for neural decoding". In: *Engineering in Medicine and Biology Society, 2004. IEMBS'04. 26th Annual International Conference of the IEEE*. Vol. 2. IEEE, pp. 4009–4012.

Wouters, Jasper, Fabian Kloosterman, and Alexander Bertrand (2018). "Towards online spike sorting for high-density neural probes using discriminative template matching with suppression of interfering spikes". In: *Journal of Neural Engineering*.

Wu, Samuel M, Fan Gao, and Bruce R Maple (2000). "Functional architecture of synapses in the inner retina: segregation of visual signals by

stratification of bipolar cell axon terminals". In: *Journal of Neuroscience* 20.12, pp. 4462–4470.

Yger, Pierre, Giulia LB Spampinato, Elric Esposito, Baptiste Lefebvre, Stéphane Deny, Christophe Gardella, Marcel Stimberg, Florian Jetter, Guenther Zeck, Serge Picaud, Jens Duebel, and Olivier Marre (2016). "Fast and accurate spike sorting in vitro and in vivo for up to thousands of electrodes". In: *bioRxiv*, p. 067843.

— (2018). "A spike sorting toolbox for up to thousands of electrodes validated with ground truth recordings in vitro and in vivo". In: *Elife* 7, e34518.

Zhang, Tian, Raghu Ramakrishnan, and Miron Livny (1996). "BIRCH: an efficient data clustering method for very large databases". In: *ACM sigmod record* 25.2, pp. 103–114.

Zhang, Yifeng, In-Jung Kim, Joshua R Sanes, and Markus Meister (2012). "The most numerous ganglion cell type of the mouse retina is a selective feature detector". In: *Proceedings of the National Academy of Sciences* 109.36, E2391–E2398.

Zipfel, Warren R, Rebecca M Williams, and Watt W Webb (2003). "Nonlinear magic: multiphoton microscopy in the biosciences". In: *Nature biotechnology* 21.11, pp. 1369–1377.

Résumé

Les neurones sont les unités de calcul fondamentales du système nerveux central. De récentes avancées technologiques permettent d'enregistrer simultanément l'activité de milliers de cellules. Le développement de réseaux de microélectrodes qui possèdent des milliers d'électrodes groupées densément en est un exemple typique. Il renouvelle le défi du tri des potentiels d'actions des signaux enregistrés. Je passe d'abord en revue les problèmes associés aux méthodes de tri, et je compare les algorithmes qui ont été proposés. Je présente ensuite un nouvel algorithme permettant de trier les potentiels d'action en ligne pour des enregistrements à grande échelle. Le partitionnement en ligne basé sur la densité et l'appariement de motifs sont essentiels pour obtenir de bonnes performances. Le logiciel a été validé sur des enregistrements synthétiques et des données réelles de vérification. Finalement, je présente une application spécifique sur la rétine où le tri des potentiels d'action en ligne pourrait être utile. Classiquement, les cellules ganglionnaires, la sortie de la rétine, sont supposées extraire des caractéristiques spécifiques de la scène visuelle telles que des augmentations ou des diminutions de luminance (cellules ON ou OFF). Cependant, le traitement de la rétine dépend du contexte visuel. En utilisant une nouvelle approche perturbative, je montre que la même cellule peut être alternativement ON ou OFF en fonction du contexte naturel. Je montre qu'un modèle de réseau neuronal convolutif ajusté aux données peut récapituler cette dépendance au contexte. Les perturbations en ligne sont donc un outil prometteur pour sonder les calculs neuronaux des systèmes sensoriels.

Mots Clés

Neurosciences computationnelles – Électrophysiologie – Tri de potentiels d'action – Rétine
Extraction de caractéristique – Dépendance au contexte

Abstract

Neurons are the fundamental computing units of the central nervous system. Recent technological advances have made it possible to simultaneously record the activity of thousands of cells. A typical example is the development of microelectrode arrays with thousands of electrodes packed with a high density. A renewed challenge is to spike sort their recorded signals, by extracting the spiking activity of each neuron. I first review the issues associated with spike sorting methods, and compare the algorithms that have been proposed. I then present a new algorithm to sort spikes online from large-scale recordings. Online density-based clustering and template matching are key to reach good performances. The software has been validated on both synthetic and real ground-truth recordings. Finally, I present a specific application on the retina where online spike sorting might be useful. Classically, ganglion cells, the retinal output, are supposed to extract specific features from the visual scene such as increases or decreases of luminance (ON vs OFF cells). However, retinal processing depends on the visual context. Using a novel perturbative approach, I show that the same cell can turn ON or OFF depending on the natural context. I found that a convolutional neural network model fitted to the data can recapitulate context-dependence. Online perturbations are thus a promising tool to probe computations in sensory systems.

Keywords

Computational neuroscience – Electrophysiology – Spike sorting – Retina
Feature extraction – Context dependency