

CHARACTERIZATION OF VISUAL CELLS USING GENERIC MODELS AND
NATURAL STIMULI

by

Joaquín Rapela

A Dissertation Presented to the
FACULTY OF THE USC GRADUATE SCHOOL
UNIVERSITY OF SOUTHERN CALIFORNIA
In Partial Fulfillment of the
Requirements for the Degree
DOCTOR OF PHILOSOPHY
(ELECTRICAL ENGINEERING)

August 2010

Copyright 2010

Joaquín Rapela

UMI Number: 3418153

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.



UMI 3418153

Copyright 2010 by ProQuest LLC.

All rights reserved. This edition of the work is protected against unauthorized copying under Title 17, United States Code.



ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

Dedication

To my family, to which I owe much of what I am.

Acknowledgements

I am specially grateful for traveling the beautiful path of my PhD in the close company of my advisor Dr. Norberto Grzywacz. He is a role model of a scientist, whose intelligence dazzles me with aesthetical pleasure. In addition, I am thankful for the company of Norberto the human being, with an endless source of warmth, optimism, encouragement, and energy.

I thank my co-advisor Dr. Jerry Mendel for his guidance, for allowing me to pursue my PhD in the beautiful area of Signal Processing, and for my office in the Electrical Engineering Building, where I interact with brilliant engineers, and from where I can admire the beautiful sunsets of Los Angeles.

Thanks to Dr. Vasilis Marmarelis for his advice in the exciting field of nonlinear modeling of physiological systems. I also thank him and Dr. Judith Hirsch for being in my thesis committee.

I am indebted to Dr. Bart Kosko for his hard advice, especially for motivating me to study advanced mathematics, and to Dr. Larry Goldstein for showing me a beautiful world of mathematics and a delightfully kind personality.

I have been fortunate to be a member of the Visual Processing Laboratory, not only an excellent place to do science, but also a unique place to meet wonderful people. Pepe Barraza, David Merwine, Susmita Chatterjee, Xiwu Cao, Eun Jin Lee, Junkwan Lee, Monica Padilla, Nadav Ivzan, Arvind Iyer, Yerina Ji, Denise Steiner, Norberto Grzywacz and Consuelo Correa, thanks to all.

Table of Contents

Dedication	ii
Acknowledgements	iii
List Of Figures	v
Abstract	xvi
Chapter 1 INTRODUCTION	1
1.1 Estimation of general non-parametric models of visual cells from their responses to natural stimuli	1
1.2 Part I - the Volterra Relevant-Space Technique (VRST)	3
1.3 Part II - the Extended Projection Pursuit Regression (ePPR) algorithm	5
1.4 The bias/variance dilemma	10
Chapter 2 THE VOLTERRA RELEVANT SPACE TECHNIQUE (VRST)	12
2.1 Introduction	13
2.2 Theory	17
2.3 Methods	25
2.4 Results	39
2.5 Discussion	72
2.6 Appendices	80
Chapter 3 THE EXTENDED PROJECTION PURSUIT REGRESSION (EPPR) ALGORITHM	87
3.1 Introduction	88
3.2 Extended Projection Pursuit Regression	92
3.3 Simulated cell	98
3.4 Complex cell	108
3.5 Simple cell	113
3.6 Discussion	116
3.7 Methods	125
3.8 Appendices	139
Chapter 4 DISCUSSION	164
References	169

List Of Figures

- 1.1 Cortical complex cell: predictions with two classes of models and two classes of estimated low-dimensional subspaces. This figure shows average correlation coefficients between cell responses and model predictions. These coefficients appear as functions of the number of inputs used to train the models. Black curve: Volterra model using low-dimensional subspaces estimated by PPR. Red curve: Volterra model using low-dimensional subspaces estimated by STC. Green curve: Nonlinearity estimated from histograms using low-dimensional subspaces estimated by STC. Blue curve: Nonlinearity estimated from histograms using low-dimensional subspaces estimated by PPR. The size of the error bars is two standard errors. For any number of inputs, the Volterra models yields better predictions than the histogram based model with both low-dimensional subspaces estimated by PPR (black versus blue curves) or by STC (red versus green curves). For more than 1,000 inputs, the PPR low-dimensional subspaces produced equal or better predictions than the STC low-dimensional subspaces, for both the Volterra (black versus red curve) or histogram-based (blue versus green curve) models. 6
- 1.2 ePPR models estimated from responses of a complex cell to natural and random data. (a,b): filters (a) and nonlinear functions (b) of a model estimated from natural data. The titles in (a) are the corresponding β coefficients. (c,d): as (a,b) but for models estimated from random data. (e): average number of terms in ePPR models estimated from natural and random data. (f-g) predictive power of filters estimated by different methods. The estimated filters and nonlinear functions are consistent with those estimated using previous methods. Models estimated from natural and random data are similar to each other. However, late suppression is only present in the model estimated from natural data. Furthermore, models estimated from natural data recovered more filters than models estimated from random data. For natural data, ePPR filters yielded predictions substantially better than those estimated by other methods, and predictions from ePPR filters were close to the upper bound on the predictive power of any model. For random data, ePPR and MID filters estimated from 20,000 stimuli were very similar and their predictions were not statistically different. 9

2.1 Sample natural images. 25

2.2 Goodness of fit of PPR models as a function of number of estimated relevant dimensions. (a) Simulated simple cell. (b) Simulated complex cell. (c) Cortical complex cell. The size of error bars is 10 standard errors. Based on these data, we use two, two, and three estimated relevant dimensions for the Volterra model of the simulated simple cell, simulated complex cell, and cortical complex cell respectively. 40

2.3 Simulated simple cell: relevant dimensions. (a), (c): analytical relevant dimension. (b), (d): projection of the analytical relevant dimension onto the estimated relevant space. (a), (b): perspective plot. (c), (d): contour plot. The analytical relevant dimension is accurately approximated by its projection onto the relevant space, $r^2 = 0.94$ 42

2.4 Correlation coefficients between cell responses and the Volterra-model predictions versus the order of the Volterra model. (a) Simulated simple cell. (b) Simulated complex cell. (c) Cortical complex cell. The size of error bars is two standard errors. The order of the Volterra model was selected as the least order maximizing the average predictive power. We selected a 3rd, 2nd, and 2nd order for the simulated simple cell, simulated complex cell, and cortical complex cell respectively. 43

2.5 Simulated simple cell: mean relative contributions of Volterra terms to the response. Contributions have been averaged over different response ranges: (a) small responses (first quartile), (b) medium responses (second and third quartiles), and (c) large responses (fourth quartile). As the responses increase, high-order Volterra terms become more important. . . 44

2.6 Simulated simple cell: first-order kernels. (a), (c): true. (b), (d) estimated. (a), (b): perspective plot. (c), (d): contour plot. Despite the noise, the estimated kernel accurately approximates its true value, $MSE=2.42E-03$. . 46

2.7 Simulated simple cell: second-order kernel slices with respect to Position (6, 6). (a), (c): true. (b), (d) estimated. (a), (b): perspective plot. (c), (d): contour plot. Blue arrows in the contour plots indicate the reference position. Despite the noise, the estimated kernel accurately approximates its true value, $MSE=1.83E-03$ 47

2.8 Simulated simple cell: comparison between PPR and STA. The figures show the r^2 statistic for the regression of the coefficients of the true and estimated relevant dimensions. This statistic appears as a function of (a) the number of inputs, when the mean response of the simulated cell is 5 spikes/image, and (b) the mean number of spikes in the responses, when using 5,000 image/response pairs. Black curve: r^2 statistic between the true relevant dimension and its projection onto a space spanned by two PPR relevant dimensions. Red curve: r^2 statistic between the true and one PPR relevant dimensions. Green curve: r^2 statistic between the true and STA relevant dimensions. The size of the error bars is two standard errors. PPR performs better than STA either when using one (red versus green curve) or two (black versus green) relevant dimensions to approximate the true relevant dimension. 49

2.9 Simulated simple cell: predictions of the Volterra model and the nonlinearity estimated from histograms. This figure shows average correlation coefficients between simulated cell responses and model predictions. These coefficients appear as functions of (a) the number of inputs (using a mean response of 5 spikes/image) and (b) the number of spikes per image (using 5,000 image/response pairs). Black curve: Volterra model using the two PPR relevant dimensions. Red curve: Volterra model using the STA relevant dimension. Green curve: Nonlinearity estimated from histograms using the STA relevant dimension. Blue curve: Nonlinearity estimated from histograms using the first PPR relevant dimension. The size of error bars is two standard errors. For sufficiently high number of inputs and signal-to-noise ratio, the Volterra model using the relevant dimensions estimated by PPR performs better than the nonlinearity estimated from histograms. When using the STA relevant dimension, both models perform equally. 50

2.10 Simulated complex cell: first relevant dimension. (a), (c): first analytical relevant dimension. (b), (d): projection of first analytical relevant dimension onto the estimated relevant space. (a), (b): perspective plot. (c), (d): contour plot. The analytical relevant dimension is well approximated by its projection onto the estimated relevant space, $r^2 = 0.96$ 52

2.11 Simulated complex cell: second relevant dimension. (a), (c): second analytical relevant dimension. (b), (d): projection of second analytical relevant dimension onto the estimated relevant space. (a), (b): perspective plot. (c), (d): contour plot. The analytical relevant dimension is well approximated by its projection onto the estimated relevant space, $r^2 = 0.95$ 53

2.12 Simulated complex cell: mean relative contributions of Volterra terms to the response. Contributions have been averaged over different response ranges: (a) small responses (first quartile), (b) medium responses (second and third quartiles), and (c) large responses (fourth quartile). Medium and large responses of the purely-quadratic complex cell model are dominated by the second order Volterra term. The contributions of the zeroth-order term reflect a spurious fit (see text). 55

2.13 Simulated complex cell: first-order kernels. (a), (c): true. (b), (d): estimated. (a), (b): perspective plot. (c), (d): contour plot. The kernels have been scaled to reflect their mean contribution to the cell response. Although the estimated but not the true kernel has some structure, the amplitude is low, making the estimation not much different from zero, MSE=5.6E-04. 57

2.14 Simulated complex cell: second-order kernel slices with respect to Position (6, 6). (a), (c): true. (b), (d): estimated. (a), (b): perspective plot. (c), (d): contour plot. Blue arrows in the contour plots indicate the reference position. Despite the noise, the estimated kernel accurately approximates its true values, MSE=1.42E-03. 58

2.15 Simulated complex cell: comparison between PPR and STC for natural images at various conditions. These conditions were the following: Red curve: STC1, equal-variance responses; Green curve: STC2, non-equal-variance responses with variance-equalization pre processing; Blue curve: STC3, non-equal-variance responses without variance-equalization pre processing. The black curve is for PPR with non-equal-variance responses. The figure shows average r^2 statistic for the two relevant dimensions of the simulated complex cell. This statistic appears as a function of (a) the number of inputs, when the mean response is 5 spikes/image, and (b) the number of spikes per image, when using 5,000 image-response pairs. The size of the error bars is two standard errors. STC achieves its best performance for responses to equal-variance images. Variance-equalization pre processing improves the performance of STC when using responses to non-equal-variance images. For responses to non-equal variance images, PPR outperforms STC for any number of inputs and signal-to-noise ratios. 60

2.16 Simulated complex cell: predictions of the Volterra model and the nonlinearity estimated from histograms. This figure shows average correlation coefficients between simulated cell responses and model predictions. These coefficients appear as functions of (a) the number of inputs when the mean responses was 5 spikes/image, and (b) the number of spikes per image, when using 5,000 image/response pairs. Black curve: Volterra model using the PPR relevant dimensions. Red curve: Volterra model using the STC relevant dimensions. Green curve: nonlinearity estimated from histograms using the STC relevant dimensions. Blue curve: Nonlinearity estimated from histograms using the PPR relevant dimensions. The size of the error bars is two standard errors. The relevant dimensions estimated by PPR yield better predictions than those estimated by STC. When using the PPR relevant dimensions, the Volterra model predicts responses slightly better than the histogram-based model. 62

2.17 Cortical complex cell: PPR relevant dimensions. (a), (d): first relevant dimension. (b), (e): second relevant dimension. (c), (f): third relevant dimension. (a), (b), (c): perspective plot. (d), (e), (f): contour plot. The relevant dimensions have clear structure, showing a 45° orientation preference. 63

2.18 Cortical complex cell: mean relative contributions of Volterra terms to the response. Contributions have been averaged over different response ranges: (a) small responses (first quartile), (b) medium responses (second and third quartiles), and (c) large responses (fourth quartile). This complex cell has a large spontaneous activity. Relative contributions from the first-order term are negligible. As the response level increases relative contributions from the spontaneous term become smaller, while relative contributions from the second-order term become larger. 64

2.19 Cortical complex cell: first and second-order kernels. (a), (c): first order. (b), (d): second-order kernel slice with respect to Position (5, 9). (a), (b): perspective plot. (c), (d): contour plot. The blue arrow in (d) points to the reference position of the second-order kernel slice. Because contributions from the first order Volterra term are negligible (Fig. 2.18), as in the case of the simulated complex cell, the first order kernel is probably spurious. The second-order kernels slice indicates that positive correlations between a point of light at the reference position and a point of light in its close surrounding will induce the strongest pairwise response facilitation. In turn, positive correlations between a point of light at the reference position and a point of light in the central diagonal (along the anti-preferred orientation) will induce weaker facilitation. 66

2.20 Cortical complex cell: second-order kernel slices estimated from PPR relevant dimensions. Slices correspond to different reference positions. (a), (c): Position (7, 7); (b), (d): Position (9, 5). Blue arrows in the contour plots indicate the reference positions. The slice with respect to Position (7, 7), left column, is nearly flat and contains the smallest pairwise facilitation among all second-order slices. For the slice with respect to Position (9, 5), right column, facilitation is strongest for interactions between the reference position and points in its close vicinity, or points in the brightest region on the top-left quadrant. Facilitation is weaker for interactions between the reference position and points along the central red band. 68

2.21 Cortical complex cell: STC relevant dimensions. (a), (c): first relevant dimension. (b), (d): second relevant dimension. (a), (b): perspective plots. (c), (d): contour plots. These relevant dimensions show a cleaner structure than that of the PPR relevant dimensions in Fig. 2.17. However, best relevant dimensions are those that lead to better predictions (see text). 69

2.22 Cortical complex cell: predictions with two classes of models and two classes of estimated relevant dimensions. This figure shows average correlation coefficients between cell responses and model predictions. These coefficients appear as functions of the number of inputs. Black curve: Volterra model using the PPR relevant dimensions. Red curve: Volterra model using the STC relevant dimensions. Green curve: Nonlinearity estimated from histograms using the STC relevant dimensions. Blue curve: Nonlinearity estimated from histograms using the PPR relevant dimension. The size of the error bars is two standard errors. For any number of inputs, the Volterra models yields better predictions than the histogram based model with both the PPR relevant dimensions (black versus blue curves) and the STC relevant dimensions (red versus green curves). For more than 1,000 inputs, the PPR relevant dimensions produced equal or better predictions than the relevant dimensions estimated by STC, for both the Volterra (black versus red curve) or histogram-based (blue versus green curve) models. 71

3.1 A generic LN model. The prediction of the model at time bin i , $\hat{y}(i)$, depends on the image presented at time bin i , plus the images presented at the previous D time bins. At each delay d , the model contains M_d filters. To generate its output, the model projects the input image at delay d on the M_d filters at that delay, generating scalars $g_{1,d}, \dots, g_{M_d,d}$. Then, the scalars at all delays, $\{g_{m,d}, 0 \leq d \leq D, 1 \leq m \leq M_d\}$, are used as inputs to a nonlinear function N , that predicts the cell's spike rate at time bin i . Because this model has too many parameters, previous methods have estimated simplified versions of this model. The red box surrounds the filters of the models estimated by Chichilnisky et al. (2001) and by Sharpee et al. (2006; 2008). The blue box surrounds the filters of the spatial models estimated by Touryan et al. (2005) and by Rapela et al. (2006). 90

3.2 Simulated cell: ePPR models. (a): filters of the simulated model (Equation 3.8). (b,c): filters (b) and nonlinear functions (c) of the example model estimated from natural data. The titles in (b) are the corresponding β coefficients. (d): principal angles between the true filters and those of models estimated from fitting subsets of the data with intermediate level of noise. (e-g): as (b-d) but for models estimated from random data. (h) average number of terms in ePPR models estimated from natural and random data. For both example models, the estimated filters are similar to the true filters, and the nonlinear functions correctly indicate the facilitatory/suppressive nature of the corresponding filters. 101

3.3 Simulated cell: example set of MID filters estimated from natural data (a) and random data (b). The filters are sorted from left to right in their estimation order. As in the other four sets of MID filters, the first filter is a good estimate, one of the other two filters (the third filter in the example) is an mediocre estimate, the remaining filter (the second filter in the example) is a poor estimate, and the suppressive filter is missing. MID well approximated the true filter space only along one dimension (see red curves in Figures 3.2d and 3.2g). 102

3.4 Simulated cell: predictive power of filters estimated by different methods. Pearson correlation coefficient between polynomial models predictions and simulated cell responses, as a function of the mean number of spikes per image, or noise level, in the simulated responses. The orange, red, blue, and cyan curves correspond to polynomials constructed with ePPR, MID, PPR, and nSTC filters, respectively. Black curves: upper bound on the correlation coefficients. (a): predictions with natural stimuli. (b): predictions with random stimuli. Light red asterisks mark number of spikes/image at which correlation coefficients for ePPR models are significantly greater than those for MID models. For both natural and random data, the polynomial models constructed with spatio-temporal filters, ePPR and MID, predict substantially better than those constructed with spatial filters, PPR and nSTC. Also, ePPR filters yielded significantly better predictions than MID filters for all conditions, excluding random data and intermediate amount of noise (0.56 spikes/sec). 104

3.5 LNL model: (a): power spectrum of the responses of the LN (black curve) and LNL (red curve) models. The lowpass filter had a large effect on the responses of the LN model. (b,c): estimated filters (b) and nonlinear functions (c) at delays 7, 8, and 9. The ePPR estimation procedure discarded the irrelevant terms from the forward model and learned the correct model structure for the simulated LNL model, with two facilitatory terms at delay 7, two facilitatory terms at delay 8, and one suppressive term at delay 9. Although the filters for the LNL model are worse estimates than those for the LN model (Figure 3.10a), the former filters still are reasonably good approximations of the true filters. 107

3.6 Complex cell: ePPR models. (a,b): filters (a) and nonlinear functions (b) of the example model estimated from natural data. The titles in (a) are the corresponding β coefficients. (c,d): as (a,b) but for models estimated from random data. (e): average number of terms in ePPR models estimated from natural and random data. (f-g) predictive power of filters estimated by different methods with same format as in Figure 3.4. The estimated filters and nonlinear functions are consistent with those estimated using previous methods. Models estimated from natural and random data are similar to each other. However, late suppression is only present in the model estimated from natural data. Furthermore, models estimated from natural data recovered more filters than models estimated from random data. For natural data, ePPR filters yielded predictions substantially better than those estimated by other methods, and predictions from ePPR filters were close to the upper bound on the predictive power of any model. For random data, ePPR and MID filters estimated from 20,000 stimuli were very similar (cf. Figures 3.6c and 3.7b) and their predictions were not statistically different. 110

3.7 Complex cell: example set of MID filters estimated from 20,000 responses. Same format as in Figure 3.3. For natural data (a), only the first filter is well structured. For random data (b), the three filters are well structured, and are very similar to those estimated by ePPR (Figure 3.6c). 111

3.8 Simple cell: ePPR models. The format of this figure is identical to that of Figure 3.6, but the cyan curves in (f,g) correspond to polynomial models constructed with rSTA filters. The figure in (e) does not contain error bars because, for each number of inputs, all estimated models had the same number of terms. The estimated filters and nonlinear functions are consistent with those estimated using previous methods. Models estimated with natural and random data are similar to each other. However, the model estimated from natural data, but not that estimated from random data, has features of a complex cell model. Furthermore, models estimated from natural data recovered more filters than those estimated from random data. For natural stimuli, predictions from ePPR filters were better or equal than those of previous methods. For random stimuli the filters estimated by the four methods gave similar predictions. 115

3.9 Selection of the best ePPR model by cross-validation. (a) correlation coefficients between ePPR model predictions and cell responses as a function of the number of terms in ePPR models. For each number of terms, n , the value j along the y -axis, $1 \leq j \leq 8$, is the correlation coefficient between the responses from the simulated cell to images in the j th validation subset, and the predictions of the ePPR model with n terms. The ePPR model with 4 terms predicts better than all models with smaller number of terms ($p < 0.05$ for all Wilcoxon signed-rank tests). We could not conclude that the ePPR models with 4 terms predicted worse than any model with more than 4 terms ($p > 0.1$ for all Wilcoxon signed-rank tests). Thus, the best ePPR model contains 4 terms. (b): filters of the ePPR model with 4 terms. The third filter from the left is spurious. For the simulated cell, spurious filters appeared only in models estimated from responses with the largest noise level. They were removed from the model using the “Removal of spurious terms” procedure. (c): final ePPR estimate obtained by applying the “Removal of spurious terms” procedure to the ePPR model with 4 terms. 131

3.10 Simulated cell: ePPR models without time interaction estimated from responses to natural stimuli. (a,b): filters (a) and nonlinear functions (b) of an example model estimated from responses to natural stimuli with intermediate amount of noise (0.56 spikes/image). The titles in (a) are the corresponding β coefficients. (c) predictive power of ePPR models with and without time interaction compared to that of a polynomial model. Orange curve: predictions for a second-order multi-dimensional polynomial constructed with ePPR filters with time interaction (re-plotted from Figure 3.4a). Red curve: predictions from ePPR models with time interaction. Pink curve: predictions from ePPR models without time interaction. Black curve: upper bound on correlation coefficients. Light red asterisks mark number of spikes/image at which predictions of the ePPR models with time interaction were significantly better than those of the polynomial models. Despite the mismatch between the simulated model in Equation 3.8, that incorporates time interactions between pixels of images at different delays, and the ePPR model without time interactions, that cannot model these interactions, the estimated filters (Figures 3.10a) well approximate the true filters (Figure 3.2a), and the estimated nonlinear functions correctly recovered the facilitatory/suppressive nature of the associated filters. 156

3.11 Complex cell: ePPR models without time interaction estimated from responses to natural stimuli. Same format at Figure 3.10. The filters and nonlinear function of this model are similar to those of the model with time interaction (Figure 3.6). The ePPR model with time interaction predicts significantly better than the ePPR without time interactions, demonstrating the relevance of nonlinear interactions between pixels of images at different delay for the response of this complex cell. 157

3.12 Varying the amount of divisive inhibition. (a): principal angles between the true filters of the simulated model (Figure 3.2a) and those of ePPR models with time interaction estimated from responses with varying amount of inhibition. (b,c): filters (b) and nonlinear functions (c) of the ePPR model estimated from responses with the largest inhibition and whose filters were most different from the true filters. Even though ePPR models cannot represent exactly the simulated model with divisive inhibition, they yields good approximations for a broad range of inhibition strengths. 159

3.13 Temporally correlated inputs. (a,b): filters (a) and nonlinear functions (b) of ePPR model without time interactions. (c,d): filter (c) and nonlinear function (d) of the ePPR model with time interactions. (e): MID filters. (f): predictive power of the different filters. Light red line with asterisks indicate that the model on the left generated better predictions than the model on the right. High correlations in the inputs substantially degrade the quality of the ePPR and MID estimates. Due to the reduced number of effective stimuli, ePPR models without time interaction yielded better estimates than ePPR models with time interaction. Also, predictions from filters of ePPR models without timer interactions were significantly better than those from filters of ePPR models with time interaction, and those from MID filters. 162

3.14 Population results: (a): correlation coefficients between mean cell responses to natural stimuli and predictions from ePPR models without time interaction, as a function of the maximal correlation coefficient between pairs of responses to repetitions of the stimuli. (b,c): filters (b) and nonlinear functions (c) of the ePPR model without time interaction estimated from responses of the cell with the lowest correlation coefficient between cell responses and models predictions. Black circles: complex cells. Blue circles: simple cells. Solid circles: example cells shown in this article. Only cells for which the maximal correlation coefficient between pairs of responses was greater than 0.1 are shown. The cortical complex cell studied in this article is the one for which we obtained best correlation coefficients, but similar correlation coefficients, and qualitatively similar ePPR estimates, were obtained for other complex cells. For example, the filters and nonlinear functions of the cell with the lowest correlation coefficient between mean cell responses and predictions from ePPR models are qualitatively similar to those of the complex cell studied in this article (Figures 15a and 15b). 163

Abstract

Traditionally visual cells have been characterized using their responses to artificial stimuli by simple parametric models. However, recent investigations show that visual cells adapt to the statistical properties of the stimuli used to probe them. Thus, to characterize visual cells in their natural operating conditions, it is important to use naturalistic stimuli. Simple parametric models are designed for specific classes of cells, making assumptions about their response properties. But, if these assumptions do not match the cell response properties, the interpretation of the estimated model parameters is questionable. An alternative is to use generic non-parametric models that can characterize a broad range of cell classes.

This thesis contains technical and scientific contributions. Technically, we develop methods to estimate generic non-parametric models of visual cells from their responses to arbitrary, including natural, stimuli. In the first part of this thesis, we introduce the Volterra Relevant Space Technique (VRST), that allows the estimation of spatial Volterra models of visual cells from their responses to natural stimuli. Disregarding temporal properties of the response generation mechanism for the estimation of spatial Volterra models is a good first approximation. However, in most conditions responses of visual cells are not spatial, but spatio temporal. So, in the second part of this dissertation we build the extended Projection Pursuit Regression (ePPR) algorithm, that estimates a very general model for the characterization of visual cells in space and time. The generality of the ePPR model reveals differences in response properties of cortical cells to natural and random stimuli that had not been observed with existing models. Thus,

scientifically this thesis shows that using natural stimuli for the characterization of visual cells is relevant to understand natural vision.

Chapter 1

INTRODUCTION

This thesis presents the material described in two self contained papers. In this introduction we describe our line of investigation (Section 1.1), summarize the results of these papers (Sections 1.2 and 1.3), which appear in the following two chapters, and conclude discussing the bias/variance dilemma, a central problem overarching the research in both papers (Section 1.4).

1.1 Estimation of general non-parametric models of visual cells from their responses to natural stimuli

Systems-level neuroscientists have a few favorite questions, the most prominent of which is the ‘what’ part of the neural coding problem: what makes a given sensory cell in a particular part of the brain fire? This question has been addressed by presenting stimuli to a cell, recording its responses, and from these stimuli and responses estimating the parameters of a model relating properties of the stimuli to properties of the cell responses. Then, answers to the what question have been inferred from the estimated model parameters (e.g., Hartline, 1940; Kuffler, 1953; Hubel & Wiesel, 1962; P. Marmarelis & Naka, 1972; P. Marmarelis & Marmarelis, 1978; Movshon, Thompson, & Tolhurst, 1978b, 1978a; Citron, Kroeker, & McCann, 1981; DeAngelis, Ohzawa, & Freeman, 1993b; Dan, Attick, & Reid, 1996; Ringach, Sapiro, & Shapely, 1997; Kaplan & Benardete, 2001;

Chichilnisky, 2001; Baccus & Meister, 2002; David, Vinje, & Gallant, 2004; Prenger, Wu, David, & Gallant, 2004; Wu, David, & Gallant, 2006; Rapela et al., 2006; Wang et al., 2007; Mante, Bonin, & Carandini, 2008; Rapela, Felsen, Touryan, Mendel, & Grzywacz, 2010).

Two key issues in this line of investigation are the type of stimuli used to probe the cell and the type of model used to represent it. Regarding the stimuli, traditionally sensory neurons have been studied with small sets of simple stimuli, specifically designed to probe certain aspects of their response properties (e.g., Hartline, 1940; Kuffler, 1953; Barlow, 1953; Liberman, 1982), or with large sets of random stimuli (e.g., Citron et al., 1981; Ringach et al., 1997; Chichilnisky, 2001; Baccus & Meister, 2002; Pillow et al., 2008). However, recent investigations show that the response of cells in the visual system (David et al., 2004; Felsen, Touryan, Han, & Dan, 2005; Sharpee et al., 2006; Wang et al., 2007; Sharpee et al., 2008), in the auditory system (Theunissen & Shaevitz, 2006; Theunissen et al., 2001; Wooley, Gill, & Theunissen, 2006), and in the somatosensory system (Maravall, Petersen, Fairhall, Arabzadeh, & Diamond, 2007) adapt to the statistical properties of the stimuli used to probe them. Thus, to characterize sensory cells in their natural operating conditions, it is important to use naturalistic stimuli.

Regarding the model, cells are normally characterized with simple parametric models, specifically designed for a given class of cells (e.g., Movshon et al., 1978b, 1978a; DeAngelis et al., 1993b; Dan et al., 1996; Kaplan & Benardete, 2001; Chichilnisky, 2001; Baccus & Meister, 2002; David et al., 2004; Felsen et al., 2005; Mante et al., 2008). To build these simple parametric models, one hypothesizes a set of rules that determine the response of a class of cells, and then uses responses of a given cell of this class to fine tune these rules for the characterization of the given cell. However, if the hypothesized set of rules are not adequate to describe responses of the cells, estimated parametric models will be suboptimal to characterize these cells, and the interpretation of their parameters will be questionable. An alternative is to use generic non-parametric models that can

characterize a broad range of cell classes (e.g., P. Marmarelis & Naka, 1972; P. Marmarelis & Marmarelis, 1978; Prenger et al., 2004; Wu et al., 2006; Rapela et al., 2006, 2010). These models avoid the tedious, difficult, and error-prone process of handcrafting the rules used to construct parametric models. Using non-parametric models allows one to “teach” computers by example so that they can “discover” the various heuristics and rules needed to characterize a given cell.

In this thesis we develop two generic non-parametric methods for the characterization of visual cells from their responses to arbitrary, including natural, stimuli. In the first part of this dissertation we introduce the Volterra Relevant Space Technique (VRST, Rapela et al., 2006, Chapter 2) that allows the estimation of spatial Volterra models, i.e., Volterra models whose response depend on only one image presented to the cell prior to its response, from responses of visual cells to natural stimuli. We summarize the VRST in Section 1.2. Disregarding temporal facets of the response generation mechanism for the estimation of spatial Volterra models is a good first approximation. But in most conditions responses of visual cells are not spatial, but spatio temporal, i.e., responses of visual cells depend on several images presented to the cell prior to its response. So, in the second part of this dissertation we present the Extended Projection Pursuit Regression algorithm (ePPR, Rapela et al., 2010, Chapter 3) for the spatio-temporal characterization of visual cells. Our results with the ePPR algorithm are summarized in Section 1.3.

1.2 Part I - the Volterra Relevant-Space Technique (VRST)

The Volterra model has a long history in the study of nonlinear physiological systems cells (see V. Z. Marmarelis, 2004, and references therein). Spatial Volterra models are multi-variable polynomials, where each variable represents a different point in space. Due to the good approximation properties of polynomials (Stone-Weierstrass Theorem, Rudin, 1976, Theorem 7.32), spatial Volterra models can characterize a broad class of cells. However, the combination of the large number of parameters in Volterra models of

visual cells, the poorly understood statistical structure of natural stimuli (Field, 1987; Rüdeman & Bialek, 1994; Simoncelli & Olshausen, 2001), and the limited size of physiological recordings, have hindered the estimation of Volterra models from responses of visual cells to natural stimuli. To overcome these limitations we use a substantiated hypothesis stating that the responses of each visual cell depend on a specially low-dimensional subspace of the image space (de Ruyter van Steveninck & Bialek, 1988).

Several methods, in different scientific disciplines, have been proposed to estimate the low-dimensional subspace (de Boer & Kuyper, 1968; de Ruyter van Steveninck & Bialek, 1988; Helland, 1988; Li, 1991; Sharpee, Rust, & Bialek, 2004; Touryan et al., 2005). However, these methods have limitations for the estimation of general low-dimensional subspaces from natural images. So in this thesis we evaluate Projection Pursuit Regression (PPR, Friedman & Stuetzle, 1981).

With the estimated low-dimensional subspace we build a low-dimensional Volterra model. We prove that, under suitable conditions on the low-dimensional subspace, a high-dimensional Volterra model can be rewritten as a low-dimensional one. Then we fit the parameters of this low-dimensional Volterra model to physiological data. Finally, from the estimated parameters of the low-dimensional Volterra model and the estimated low-dimensional subspace, we reconstruct the parameters of the high-dimensional Volterra model. Thus, this procedure makes possible the estimation of the parameters of high-dimensional Volterra models from limited amounts of physiological data.

We first evaluate the feasibility of the above procedure with simulated data from cortical cells. We show that PPR and the VRST recover good approximations of the true low-dimensional subspaces and true Volterra models of simulated cells, which are more accurate than those obtained by previous methods. However, many algorithms perform well with simulated data, but poorly with real data. So, next we evaluate PPR and the VRST with physiological data from a cortical complex cell (Felsen et al., 2005). We compare PPR with Spike Triggered Covariance (STC, Touryan et al., 2005) for the estimation of the low-dimensional subspace, and Volterra models with a histogram-based

model proposed by Touryan et al. (2005) to predict responses from the low-dimensional subspace. Figure 1.1 is a representative example of this comparison. It plots the correlation coefficient between predictions of models and cell responses, as a function of the number of inputs used to estimate the parameters of the models. For any number of inputs Volterra models yield better predictions than histogram-based models, with both the low-dimensional subspace estimated by PPR (black vs. blue curves) and the low-dimensional subspace estimated by STC (red vs. green curves). For more than 1,000 inputs, the low-dimensional subspace estimated by PPR produced equal or better predictions than the low-dimensional subspace estimated by STC, for both the Volterra (black vs. red curve) and histogram-based (blue vs. green curve) models. Hence, of the models tested, the best combination is to use Volterra models with low-dimensional subspaces estimated by PPR.

1.3 Part II - the Extended Projection Pursuit Regression (ePPR) algorithm

In Chapter 2 we show that PPR is a good algorithm for the estimation of spatial low-dimensional subspaces of visual cells. However, for the estimation of spatio-temporal low-dimensional subspaces PPR is affected by the curse of dimensionality and, even for of spatial low-dimensional subspaces, PPR has limitations. To allow the estimation of spatio-temporal low-dimensional subspaces, and to overcome these limitations, we introduce the Extended Projection Pursuit Regression (ePPR) algorithm.

The ePPR model represents the cell response as a sum of several terms. Each terms computes the inner product between the stimuli and a filter, it uses the output of this inner product as input to a nonlinear function, and scales the output of this nonlinear function by an importance factor β . A unique feature of the ePPR algorithm is the efficient optimization algorithm that it uses to non-parametrically estimate the large number of parameters in the ePPR model. Thanks to this optimization algorithm, currently ePPR

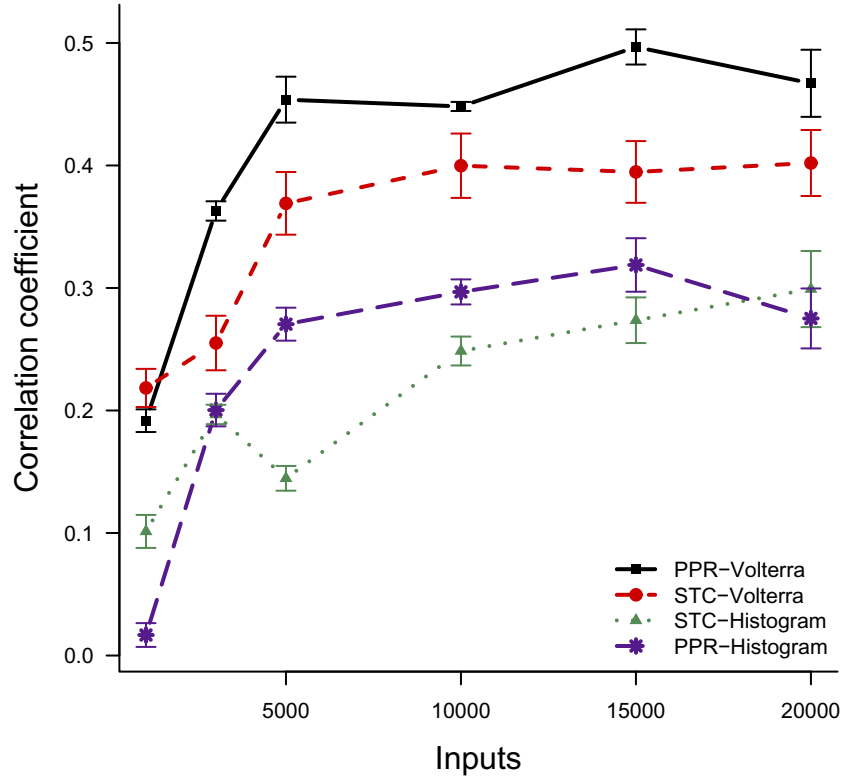


Figure 1.1: Cortical complex cell: predictions with two classes of models and two classes of estimated low-dimensional subspaces. This figure shows average correlation coefficients between cell responses and model predictions. These coefficients appear as functions of the number of inputs used to train the models. Black curve: Volterra model using low-dimensional subspaces estimated by PPR. Red curve: Volterra model using low-dimensional subspaces estimated by STC. Green curve: Nonlinearity estimated from histograms using low-dimensional subspaces estimated by STC. Blue curve: Nonlinearity estimated from histograms using low-dimensional subspaces estimated by PPR. The size of the error bars is two standard errors. For any number of inputs, the Volterra models yields better predictions than the histogram based model with both low-dimensional subspaces estimated by PPR (black versus blue curves) or by STC (red versus green curves). For more than 1,000 inputs, the PPR low-dimensional subspaces produced equal or better predictions than the STC low-dimensional subspaces, for both the Volterra (black versus red curve) or histogram-based (blue versus green curve) models.

is the only algorithm able to estimate, from responses of visual cells to natural stimuli, low-dimensional subspaces that are spatio-temporal and contain multiple filters.

Because ePPR is an entirely non-parametric algorithm, and because the cell responses are very noisy, ePPR estimates could have been very variable. However, this was not the case. Several features of the ePPR estimation procedure help reduce the variability of the estimated parameters. First, the projection pursuit strategy used by ePPR reduces the original large-dimensional problem of estimating all the parameters in the ePPR model to a sequence of low-dimensional problems. Second, the estimated filters are penalized for lack of smoothness. And third, the estimation of the nonlinear functions is performed using smoothing splines with a relatively large penalty for non-smooth estimates.

ePPR is a very general method for the characterization of visual cells. In Chapter 3 we prove that ePPR models can uniformly approximate, to an arbitrary degree of precision, any continuous function. To test this generality empirically, we show that ePPR recovers very good approximations to the parameters of models of visual cells that cannot be represented exactly with an ePPR model. Next we evaluate ePPR with physiological data from primary visual cortex, and show that it can characterize both simple and complex cells, from their responses to both natural and random stimuli.

Figure 1.2, shows an example of the features that ePPR reveals in the responses of cortical cells. It shows ePPR models estimated from responses of a real complex cell (Felsen et al., 2005) to natural and random stimuli. The spatio-temporal filters estimated from natural data (Figure 1.2a) are consistent with previous estimations of linear subspaces of complex cells from responses to natural (Touryan et al., 2005; Rapela et al., 2006), or random (Movshon et al., 1978a; Chen, Han, Poo, & Dan, 2007) stimuli. In particular, the three middle filters have clear Gabor shapes with similar orientation and spatial frequency, but are shifted in phase. However, note that the bottom frame of the rightmost filter (operating on the image presented between 85 and 126ms prior to the cell response) is cross-oriented with respect to the other filters. Figure 1.2b shows the nonlinear functions

of the model estimated from natural data. The leftmost nonlinear function is approximately a half-wave rectification. The three middle nonlinear functions, corresponding to the filters with clear Gabor shapes in Figure 1.2a, are full-squared, in agreement with the polarity invariance of complex cells (Movshon et al., 1978a). And the rightmost nonlinear function, corresponding to the filter with a cross-oriented frame at long delays, is suppressive, revealing cross-oriented inhibition in the response of this complex cell. We compared the predictive power of filters estimated by ePPR, and by previous methods: MID (Sharpee et al., 2004), nSTC (Touryan et al., 2005), and PPR (Friedman & Stuetzle, 1981), using a second-order multi-dimensional polynomial as the predictive method. Figure 1.2f plots the correlation coefficients between the complex cell responses and the polynomial models predictions, as a function of the number of stimuli used to estimate the filters and polynomial models. The dashed line is an upper bound on these correlation coefficients. For all number of stimuli best predictions are obtained with ePPR filters. Moreover, ePPR filters estimated using 20,000 inputs closely approximate the upper bound on the correlation coefficients.

The ePPR model estimated from responses of the complex cell to random stimuli (Figures 1.2c and 1.2d) is very consistent with that estimated from responses to natural stimuli (Figures 1.2a and 1.2b). The estimated filters (Figure 1.2c) are a subset of those obtained with natural stimuli (Figure 1.2a), and the nonlinear functions (Figure 1.2d) correspond to those obtained from natural stimuli (Figure 1.2b). Despite this consistency, two important differences emerged between the models estimated from natural and random data: First, a filter with late suppression was recovered from natural, but not from random, data. Second: models estimated from natural data had more filters than those estimated from random data (Figure 6g). As for natural stimuli, predictions from ePPR models estimated from responses to random stimuli are superior or equal than those of previous models (Figure 1.2g).

In summary, Figure 1.2 shows that estimated ePPR models displayed several features in common with previous characterizations of complex cells, that ePPR models estimated

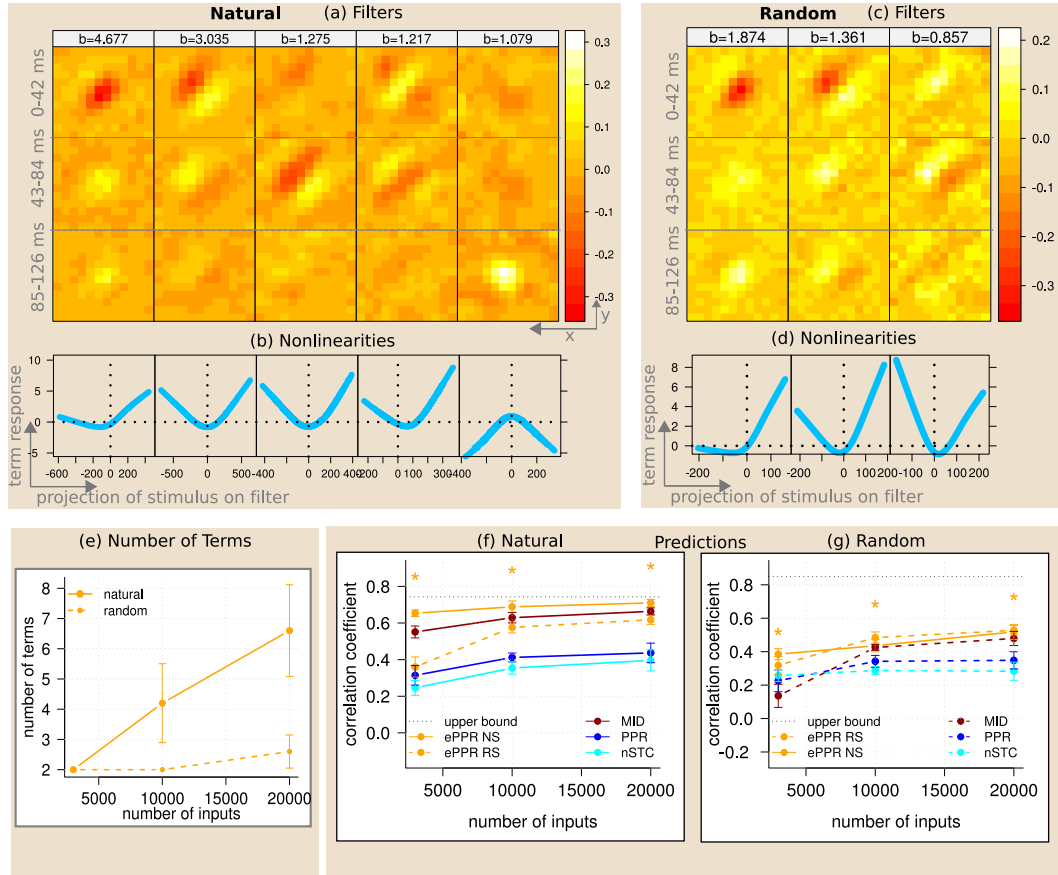


Figure 1.2: ePPR models estimated from responses of a complex cell to natural and random data. (a,b): filters (a) and nonlinear functions (b) of a model estimated from natural data. The titles in (a) are the corresponding β coefficients. (c,d): as (a,b) but for models estimated from random data. (e): average number of terms in ePPR models estimated from natural and random data. (f-g) predictive power of filters estimated by different methods. The estimated filters and nonlinear functions are consistent with those estimated using previous methods. Models estimated from natural and random data are similar to each other. However, late suppression is only present in the model estimated from natural data. Furthermore, models estimated from natural data recovered more filters than models estimated from random data. For natural data, ePPR filters yielded predictions substantially better than those estimated by other methods, and predictions from ePPR filters were close to the upper bound on the predictive power of any model. For random data, ePPR and MID filters estimated from 20,000 stimuli were very similar and their predictions were not statistically different.

from natural and random stimuli were very similar to each other, but displayed a few interesting differences, and that predictions from ePPR models were close to an upper bound on the predictive power of any model.

1.4 The bias/variance dilemma

The essence of the bias/variance dilemma lies in the fact that the mean square error of any regression model with finite variance can be decomposed into a bias and a variance component; whereas incorrect parametric models lead to high bias, very general non-parametric models suffer from high variance. Prohibitively large amounts of inputs are then needed to estimate non-parametric models. The only way to control their variance is to use less general models. However, and this is the other face of the dilemma, as we reduce the generality of a model, we increase the chances that their estimates are suboptimal, or biased (Geman, Bienenstock, & Doursat, 1992).

The bias/variance dilemma is particularly relevant for the non-parametric characterization of visual cells using natural stimuli. This happens because natural images are complex (Field, 1987; Ruderman & Bialek, 1994; Simoncelli & Olshausen, 2001), so the number of descriptors needed to represent them is large, and a generic non-parametric model, using natural images as inputs, would need a very large number of parameters. The amount of data required to estimate the parameters of a model grows exponentially with the number of model parameters. Thus, a prohibitively large amount of data—unattainable in standard physiology experiments—would be required to estimate the parameters of generic models of visual cells using natural stimuli as inputs.

This bias/variance dilemma can be circumvented by purposefully reducing the generality of non-parametric models, i.e., introducing bias. Of course, one must ensure that the bias is harmless, in the sense that the functions that the less general model can well approximate include an anticipated class of functions that one wishes to characterize. In essence, bias needs to be designed for each particular problem.

To address the bias/variance dilemma, in this thesis we make use the hypothesis that the response of each visual cells depends on a low-dimensional subspace of the image space. We then bias the estimated models to contain a small number of filters. To further reduce variability, these filters are estimated using the projections pursuit strategy, which is specifically designed to address the variability problem in the estimation of its parameters.

The rest of this dissertation is organized as follows. Chapter 2 describes the VRST, Chapter 3 presents the ePPR algorithm, and Chapter 4 closes this dissertation discussing overarching topics and outlining directions for future work.

Chapter 2

THE VOLTERRA RELEVANT SPACE TECHNIQUE (VRST)

The response of visual cells is a *nonlinear* function of their stimuli. In addition, an increasing amount of evidence is showing that visual cells are optimized to process *natural images*. Hence, finding good nonlinear models to characterize visual cells using natural stimuli is important. The Volterra model is an appealing nonlinear model for visual cells. However, their large number of parameters, and the limited size of physiological recordings has hindered its application. Recently, a substantiated hypothesis that the responses of each visual cell could depend on a specially low-dimensional subspace of the image space has been proposed. We use this low-dimensional subspace in the *Volterra relevant-space technique* to allow the estimation of high-order Volterra models. Most laboratories characterize the response of visual cells as a nonlinear function on the low-dimensional subspace. They estimate this nonlinear function using histograms and fitting parametric functions to them. Here we compare the Volterra model with these histogram-based techniques. We use simulated data from cortical simple cells, and simulated and physiological data from cortical complex cells. Volterra models yields equal or superior predictive power in all conditions studied. Several methods have been proposed to estimate the low-dimensional subspace. In this paper we test Projection Pursuit Regression (PPR), a nonlinear-regression algorithm. We compare PPR with two popular models used in vision, spike-triggered average (STA) and spike-triggered covariance (STC). We observe that PPR has advantages over these alternative algorithms. Hence, we conclude

that PPR is a viable algorithm to recover the relevant subspace from natural images and that the Volterra model, estimated through the *Volterra relevant-space technique*, is a compelling alternative to histogram-based techniques.

2.1 Introduction

Nonlinearities are ubiquitous in the response of visual cells. Just in the primary visual cortex, we find rectification (Movshon et al., 1978b; Albrecht & Valois, 1981), saturation (Maffei & Fiorentini, 1973; Dean, 1981), expansion (Albrecht & Hamilton, 1982; Sclar & Freeman, 1982), phase advance (Dean & Tolhurst, 1986; Carandini & Heeger, 1994), and cross-orientation inhibition (Morrone, Burr, & Maffei, 1982; Bonds, 1989), to mention a few. Therefore, several investigators have looked for general methods to study these nonlinearities from the retina to the cortex. In theory, white noise could be an ideal stimulus to estimate nonlinear properties. This is because if a nonlinear system is stimulated with a white-noise stimulus ensemble for a long enough time, then there is a finite probability that any given stimulus will appear, probing the nonlinear system thoroughly and efficiently. However, this rationale is weak for the visual system since the possible dimensions of stimuli space are too large (*i.e.*, the number of possible images is immense). Moreover, data are noisy and limited in quantity in physiological recordings. Hence, white noise does not provide enough signal-to-noise ratio to obtain accurate estimates of responses along all stimulus dimensions. Finally, some sensory neurons do not respond well to white noise. Thus, it would be better to use natural images to concentrate the power of the stimuli on the normal operating range of the cell.

More than 40 year ago, Barlow (1961) introduced the hypothesis that visual cells are optimized to process natural stimuli. Since then, several investigators have provided support to this natural-adaptation hypothesis (Srinivasan, Laughlin, & Dubs, 1982; Dan et al., 1996; Simoncelli & Olshausen, 2001) and have shown that natural images emphasize features of responses not prominent when using synthetic stimuli (David et al., 2004).

However, only recently, a number of studies have begun to use nonlinear models and natural stimuli to characterize the response of visual cells. These studies obtained interesting findings, but the methods had limitations for investigators interested in general receptive fields. Some groups did not attempt to map receptive fields (Dan et al., 1996; Vinje & Gallant, 2000, 2002; Weliky, Fiser, Hunt, & Wagner, 2003; Aggelopoulos, Franco, & Rolls, 2005; Guo, Robertson, Mahmoodi, & Young, 2005), while others limited their studies to linear (Theunissen et al., 2001; Ringach, Hawken, & Shapley, 2002; Smyth, Willmore, Baker, Thompson, & Tolhurst, 2003; Willmore & Smyth, 2003) or second-order nonlinear (Felsen et al., 2005; Touryan et al., 2005) receptive fields. More general nonlinearities were first studied by modeling cell responses as a linear filter followed by a point nonlinearity (Chichilnisky, 2001; Nykamp & Ringach, 2002) or by fitting a priori models (David et al., 2004). The former models cannot capture nonlinear interactions between subregions of the receptive field and, thus, might not be sufficiently general to characterize responses of large classes of cells. Later, two methods that make no assumptions on the response generation mechanism were introduced (Prenger et al., 2004; Sharpee et al., 2004). These methods are powerful but they require delicate nonlinear optimizations that could be overwhelming to many investigators.

In an attempt to overcome some of the above limitations, we explored the Volterra model (V. Z. Marmarelis, 2004). Because it does not rely on specific assumptions about the response-generation mechanism, this model can be applied to a broad class of cells. Moreover, the Volterra model has a long history in the study of nonlinear physiological systems. This model could be especially useful for visual cells, since its first-order kernels are just like standard receptive fields. Therefore, the model can generalize them to the nonlinear domain. The large number of parameters in Volterra models of visual cells and limited physiological recordings have hindered its application to the visual system. In this paper, we overcome this limitation using a recent substantiated hypothesis that the responses of each visual cell depend on a specially low-dimensional subspace of the image space (de Ruyter van Steveninck & Bialek, 1988; Brenner, Bialek, & de Ruyter van

Steveninck, 2000; Sharpee et al., 2004, 2006; Bialek & de Ruyter van Steveninck, 2005; Rust, Schwartz, Movshon, & Simoncelli, 2005). Accordingly, we introduce the *Volterra relevant-space technique* to estimate Volterra models of visual cells from natural images.

Most laboratories characterize the response of visual cells as a nonlinear function on a one- or two-dimensional subspace of the image space (Chichilnisky, 2001; Touryan, Lau, & Dan, 2002; Sharpee et al., 2004, 2006; Simoncelli, Paninski, Pillow, & Schwartz, 2004; Touryan et al., 2005; Felsen et al., 2005). To estimate this nonlinear function, they construct one- or two-dimensional histograms of the response as a function of the projection of the input images on the basis of this subspace. Then, they fit parametric functions to this histogram. Using simulated data from cortical simple cells, and simulated and physiological data from cortical complex cells, we compare the Volterra model with histogram-based techniques for simple- (Chichilnisky, 2001) and complex-cells (Touryan et al., 2005).

Several methods, in different scientific disciplines, have been proposed to estimate the low-dimensional subspace (de Boer & Kuyper, 1968; Friedman & Stuetzle, 1981; Helland, 1988; Li, 1991; de Ruyter van Steveninck & Bialek, 1988; Sharpee et al., 2004; Touryan et al., 2005). In this paper we test Projection Pursuit Regression (PPR) (Friedman & Stuetzle, 1981), a nonlinear-regression algorithm. We compare PPR with two popular models used in vision, spike-triggered average (STA) (de Boer & Kuyper, 1968) and the modification of spike-triggered covariance (STC) for natural images (Touryan et al., 2005).

We organize the rest of this article as follows: The next section (“Theory”) describes the theoretical underpinnings of the modeling of cells’ receptive fields. This section covers the Volterra model (Subsection 2.2.1) and the Volterra relevant-space technique (Subsection 2.2.2). This theoretical section is followed by Methods (Section 2.3), Results (Section 2.4), and Discussion (Section 2.5). Each theoretical material starts with an intuitive description of ideas, so that the reader can get the gist of the paper even if skipping its

mathematics. Then, those sections present the main mathematical results, leaving proofs and developments to appendices.

2.2 Theory

2.2.1 Volterra Model

A useful way to think of the Volterra model is that it is a sequence of approximations. They are given by a sequence of filters, called kernels, with which to process the input. When Volterra models are applied to visual cells, the zeroth-order kernel represents the activity when the input is absent. The first-order Volterra kernel is the linear component of the response. For example, in the spatial domain, this kernel gives the response to small pulses of input at every position. Therefore, this kernel is a good representation of what people traditionally call the receptive field of a cell. The second-order kernel represents the weight that nonlinear interactions between two inputs (*e.g.*, at two times or two positions) have on the response. In general, the r th-order kernel represents the weight that nonlinear interactions between r inputs have on the response.

The development of Volterra models relies on the mathematical notion of the Volterra series introduced by Volterra (1930). In the time domain, the output $y(t)$ of a stationary, stable, causal system can be expressed in terms of its input signal $x(t)$ through its Volterra series expansion as

$$\begin{aligned} y(t) &= k_0 + \int_0^\infty k_1(\tau)x(t-\tau)d\tau \\ &+ \int_0^\infty \int_0^\infty k_2(\tau_1, \tau_2)x(t-\tau_1)x(t-\tau_2)d\tau_1d\tau_2 \\ &+ \int_0^\infty \int_0^\infty \int_0^\infty k_3(\tau_1, \tau_2, \tau_3)x(t-\tau_1)x(t-\tau_2)x(t-\tau_3)d\tau_1d\tau_2d\tau_3 \\ &+ \dots \end{aligned} \tag{2.1}$$

A Volterra model is a truncated Volterra series, *i.e.*

$$\begin{aligned}
y_{Q,k}(t) &= k_0 + \int_0^\infty k_1(\tau)x(t-\tau)d\tau \\
&+ \int_0^\infty \int_0^\infty k_2(\tau_1, \tau_2)x(t-\tau_1)x(t-\tau_2)d\tau_1d\tau_2 + \dots \\
&+ \int_0^\infty \dots \int_0^\infty k_Q(\tau_1, \dots, \tau_Q)x(t-\tau_1)\dots x(t-\tau_Q)d\tau_1\dots d\tau_Q + \varepsilon
\end{aligned} \tag{2.2}$$

where the subscript Q in $y_{Q,k}$ indicates that the Volterra series has been truncated at order Q , the subscript k reflects the dependence of the Volterra model on the particular choice of kernels $\{k_0, \mathbf{k}_1, \dots, \mathbf{k}_Q\}$, and ε stands for possible errors due to truncation and for noise in the data.

The Volterra model given in (2.2) uses one-dimensional (1D), continuous inputs. However the input to visual cells is multi-dimensional (one dimension of time, two dimensions of space, three dimensions of color, and so on) and discrete data is normally used to characterize their responses. Here, as a first approximation, we only use the spatial dimensions of the stimuli to estimate the response of visual cells, disregarding temporal interactions. As discussed in Section 2.5, the Volterra relevant-space technique can be extended to consider spatio-temporal interactions. The Volterra series used henceforth is

$$\begin{aligned}
y_{Q,k}(\mathbf{x}) &= k_0 + \sum_{i,j=1}^N k_1(i,j)x(i,j) \\
&+ \sum_{i_1,j_1,i_2,j_2=1}^N k_2(i_1,j_1,i_2,j_2)x(i_1,j_1)x(i_2,j_2) + \dots \\
&+ \sum_{i_1,j_1,\dots,i_Q,j_Q=1}^N k_Q(i_1,j_1,\dots,i_Q,j_Q)x(i_1,j_1)\dots x(i_Q,j_Q) + \varepsilon
\end{aligned} \tag{2.3}$$

where $\mathbf{x} \in \mathbb{R}^{N \times N}$ is the input image. Any Volterra model can be represented with kernels \mathbf{k}_r that are symmetric with respect to permutations in pairs of indices (i_p, j_p) . Hence, the number of parameters in Volterra kernels is less than their number of kernel coefficients.

For example, for \mathbf{k}_2 we have the symmetry $k_2(i_1, j_1, i_2, j_2) = k_2(i_2, j_2, i_1, j_1)$, which implies that the number of parameters is $\frac{(N^2+1)N^2}{2}$ while the number of kernel coefficients is N^4 . The number of parameters of the Volterra model in (2.3), n_V , is

$$n_V = \frac{(N^2 + Q)!}{N^2! Q!} \quad (2.4)$$

For example, a fourth-order Volterra model, $Q = 4$, using images of size 16×16 as input, contains $n_V = 186,043,585$ parameters. From the limited data that can be recorded in physiological experiments, we cannot estimate such a large number of parameters. The Volterra relevant-space technique described in the next section addresses this problem.

2.2.2 The Volterra Relevant-Space Technique

Fortunately, one can reduce the dimensionality of Volterra models. This is because the response of each visual cell depends on a low-dimensional subspace of the image space. One reason for this low dimensionality is the high degree of correlation in natural images (Ruderman & Bialek, 1994; Field, 1987; Srinivasan et al., 1982). Another reason is that each neuron responds to particular features of the image, such as an edge at a given position with a given orientation. Here, we show how to use this low-dimensional subspace to build a low-dimensional Volterra model, and we prove the equivalence between the original and low-dimensional Volterra models. Thus, we establish, for the first time, a mathematical link between the low-dimensional subspace and Volterra models.

The construction of the low-dimensional Volterra model begins by projecting the input images onto a low-dimensional subspace \mathcal{S} . If \mathbf{x} is an image and $\mathcal{B} = \{\mathbf{b}_1, \dots, \mathbf{b}_L : \mathbf{b}_l \in \mathbb{R}^{N \times N}\}$ is a set of orthonormal basis functions spanning the space \mathcal{S} , then the projected image is

$$\Pi_{\mathcal{S}}(\mathbf{x}) = \sum_{l=1}^L \alpha_l(\mathbf{x}) \mathbf{b}_l \quad (2.5)$$

where

$$\alpha_l(\mathbf{x}) = \sum_{i,j=1}^N b_l(i,j) x(i,j) \quad (2.6)$$

is the projection coefficient of image \mathbf{x} onto the basis function \mathbf{b}_l . Using these coefficients, the low-dimensional Volterra model is

$$\begin{aligned}
y_{Q,k}(\mathbf{x}) &= q_0 + \sum_{l=1}^L q_1(l)\alpha_l(\mathbf{x}) + \dots \\
&+ \sum_{l_1, \dots, l_Q=1}^L q_Q(l_1, \dots, l_Q)\alpha_{l_1}(\mathbf{x}) \dots \alpha_{l_Q}(\mathbf{x}) + \varepsilon
\end{aligned} \tag{2.7}$$

What are the conditions on the space \mathcal{S} for the low-dimensional Volterra model to be an equivalent representation of the original Volterra model in (2.3)? The answer is given by Proposition 1 (Appendix 2.6.1). It states that *if a vector space \mathcal{S} is such that the response of the cell to any image equals the response to the projection of the image onto \mathcal{S} , then the original Volterra model can be rewritten as the low-dimensional Volterra model*. A vector space \mathcal{S} satisfying this condition is called a *relevant space*. That is, \mathcal{S} is a relevant space if and only if

$$y_{Q,k}(\mathbf{x}) = y_{Q,k}(\Pi_{\mathcal{S}}(\mathbf{x})) \quad \forall \mathbf{x} \in X, \tag{2.8}$$

Any set of orthonormal basis function $\mathcal{B} = \{\mathbf{b}_1, \dots, \mathbf{b}_L : \mathbf{b}_l \in \mathbb{R}^{N \times N}\}$ spanning a relevant space is called a set of *relevant dimensions*.

We are not seeking a relevant space such that the represented images look similar to the original ones. Instead, we are seeking a relevant space such that the response to the original images equals the response to the represented images. This equality is as in (2.8). Therefore, for our purposes, a relevant-space representation does not depend only on the statistics of natural images, as in the case of image reconstruction. This representation depends also on the properties of the cell under study.

How much do we gain by using the relevant space to represent Volterra models? The number of parameters in the low-dimensional Volterra model, n_{LV} , is

$$n_{LV} = \frac{(L+Q)!}{L! Q!} \quad (2.9)$$

As an example, for $L = 10$ relevant dimensions, a fourth-order ($Q = 4$) low-dimensional Volterra model will contain $n_{LV} = 1,001$ parameters. In contrast, the number of parameters in the original Volterra model for a 16×16 image is $n_V = 186,043,585$, (2.4). Therefore, if a small relevant subspace can be identified, the Volterra relevant-space technique achieves enormous dimensionality reduction.

How are the coefficients, q_i , of the low-dimensional Volterra model, (2.7), estimated? We seek coefficients that minimize the difference between true responses and those predicted by the Volterra model. To obtain a convenient method for this minimization, we begin by expressing (2.7) as the linear Volterra equation

$$\mathbf{y} = \mathbf{A}\mathbf{q} + \varepsilon \quad (2.10)$$

where, for K image/response pairs and M parameters, $\mathbf{y} \in \mathbb{R}^K$ is the response vector whose i^{th} element, $y(\mathbf{x}_i)$, is the response of the cell to the i^{th} image, $\mathbf{A} \in \mathbb{R}^{K \times M}$ is the data matrix, $\mathbf{q} \in \mathbb{R}^M$ is the vector of parameters, and $\varepsilon \in \mathbb{R}$ accounts for possible errors due to truncation and to noise in the data. For instance, for a second-order cell ($Q = 2$), the i^{th} row of matrix \mathbf{A} , \mathbf{a}'_i , is

$$\begin{aligned} \mathbf{a}'_i = [1, \alpha_1(\mathbf{x}_i), \alpha_2(\mathbf{x}_i), \dots, \alpha_L(\mathbf{x}_i), \alpha_1(\mathbf{x}_i)^2, \alpha_1(\mathbf{x}_i)\alpha_2(\mathbf{x}_i), \dots, \alpha_1(\mathbf{x}_i)\alpha_L(\mathbf{x}_i), \\ \alpha_2(\mathbf{x}_i)^2, \alpha_2(\mathbf{x}_i)\alpha_3(\mathbf{x}_i), \dots, \alpha_{L-1}(\mathbf{x}_i)\alpha_L(\mathbf{x}_i), \alpha_L(\mathbf{x}_i)^2], \end{aligned} \quad (2.11)$$

and the parameters-vector \mathbf{q} is

$$\mathbf{q} = [q_0, q_1(1), \dots, q_1(L), q_2(1, 1), q_2(1, 2), \dots, q_2(1, L), \\ q_2(2, 2), q_2(2, 3), \dots, q_2(L-1, L), q_2(L, L)]'. \quad (2.12)$$

By using the mean-square error

$$\text{MSE}(k_0, \mathbf{k}_1, \dots, \mathbf{k}_Q) = \frac{1}{n} \sum_{i=1}^n (y(\mathbf{x}_i) - y_{Q,k}(\mathbf{x}_i))^2 \quad (2.13)$$

as the goodness-of-fit criterion, we can apply any linear technique to estimate the parameters-vector \mathbf{q} . The use of linear techniques in the estimation of Volterra models removes the whiteness requirement of cross-correlation techniques. Therefore, we can use natural images to estimate Volterra models, without the need to whiten them or modifying image statistics in any way.

Having estimated the coefficients of the low-dimensional Volterra model, one approximates the true kernels, k_i , in (2.3), with

$$\hat{k}_0 = q_0 \quad (2.14)$$

$$\hat{k}_1(i, j) = \sum_{l=1}^L q_1(l) b_l(i, j) \quad (2.15)$$

...

$$\hat{k}_Q(i_1, j_1, \dots, i_Q, j_Q) = \sum_{l_1, \dots, l_Q=1}^L q_Q(l_1, \dots, l_Q) b_{l_1}(i_1, j_1) \dots b_{l_Q}(i_Q, j_Q), \quad (2.16)$$

Proposition 2 (Appendix 2.6.1) provides justification for these equations. This proposition proves that, if a cell can be represented by a Volterra model using true kernels \mathbf{k}_i , then, for any set of true relevant dimensions $\{b_i\}$ (2.8), there exists coefficients $q_0, \mathbf{q}_1, \dots, \mathbf{q}_Q$, such that a Volterra model using the true kernels, \mathbf{k}_i , generates the same

responses as a Volterra model using the projected kernels in (2.14)-(2.16). Thus, from a predictive perspective, these reconstructed kernels are indistinguishable from the true kernels.

The use of (2.7)-(2.16) assumes a known order of the Volterra model, Q . Here, we selected the order Q by cross-validation, as indicated in Section 2.3.4.1. In summary, the Volterra relevant-space technique uses the following procedure to estimate Volterra models from natural images:

1. Estimate the relevant space.
2. Project the images in the estimated relevant dimensions (2.6), obtaining the coefficients α_l .
3. Use the coefficients α_l to construct the data matrix (2.11) of the linear Volterra equation (2.10).
4. Estimate the coefficients vector, q , of the linear Volterra equation by minimizing the MSE in (2.13).
5. Reconstruct the Volterra kernels using (2.14)-(2.16).

Code implementing this procedure along with the simulated data used in Section 2.4 can be downloaded from <http://vpl.usc.edu/projects/nonlinearReceptiveFields/>.

After the relevant dimensions have been estimated, the remaining steps in the procedure (Steps 2-5) involve only linear operations, and are therefore, fast and easy to implement. In contrast, most implementations of histogram-based methods fit a nonlinear function to an estimated histogram. The selection of the optimal histogram bin size is an open question and the nonlinear fit is slower and comparatively more difficult to implement than the linear operations required by the Volterra relevant-space technique.



Figure 2.1: Sample natural images.

2.3 Methods

2.3.1 Simulated Data

Responses of the cortical simple- and complex-cell models described below were generated using 10×10 image patches extracted from calibrated natural images (van Hateren & van der Schaaf, 1998). Sample natural images are shown in Fig.2.1.

2.3.1.1 Simulated Simple Cell

The simple-cell data was generated by a model similar to those previously used to describe V1 simple cells (J. Jones, Stepnoski, & Palmer, 1987). The model consisted of a linear spatial filter followed by a sigmoidal rectification ($rect(v) = saturation / (1 + \exp(-slope * (v - threshold)))$) and a Poisson spike generator. We used a sigmoidal instead of a half-way rectification, because we wanted to use a model that was differentiable at every point.

This differentiability allowed derivation of true Volterra kernels (see Appendix 2.6.2). The simple-cell model is then given by

$$y(\mathbf{x}) = \text{rect}(\langle \mathbf{f}, \mathbf{x} \rangle) \quad (2.17)$$

where $\langle ., . \rangle$ represents the Euclidean inner-product operation and \mathbf{f} is the spatial filter. For the spatial filter we used a two-dimensional (2D) Gabor function. It had a preferred orientation of 45° from vertical, a preferred spatial frequency of 2 cycles per receptive field, a spatial bandwidth of 1.6 octaves, and an even phase. For the sigmoidal rectification of the filtered image, we used *slope*=5.0 and *threshold*=1.0, and varied the *saturation* parameter (setting it to 458, 276, 92, and 46) to obtain different mean number of spikes per image (5, 3, 1, and 0.5 respectively). Because the responses of the simulated simple cell followed a Poisson distribution, the signal-to-noise ratio in the responses increased monotonically as a function of the mean number of spikes per image.

To assess the quality of the estimated relevant dimensions, we first derived analytical relevant dimensions for the simple cell. We then computed their projection onto the estimated relevant space. If the estimated relevant space embodies the true relevant space, then the analytical relevant dimensions should equal their projection onto the estimated space. In deriving the analytical relevant dimensions, we begin by observing that the first stage of the simple-cell model is a linear filter. Hence, the response to any image will equal (up to the noise level) the response to the image projected in the span of the linear filter. Consequently, the linear filter spans the relevant space. After rescaling this filter to unit norm, it is the only relevant dimension for the simulated simple cell.

The derivatives of all orders of the simple-cell model in (2.17) are non-zero. Hence, according to (2.39), every kernel in (2.3) is non-zero. This may cause a problem when the projections of the input images onto the filter \mathbf{f} (the coefficients $\alpha_l(\mathbf{x})$ in (2.7)) are large. In this case, truncation errors will become significant. However, in the opposite case,

i.e., when these projections are small, the problem is not serious. By using a model with kernels of sufficiently high order, these truncation errors will be negligible. To minimize the effect of truncation errors, the amplitude of the filter \mathbf{f} was selected in such a way that the absolute value of the largest filtered image was 1.

2.3.1.2 Simulated Complex Cell

The complex-cell data was generated by means of a spatial energy model (Adelson & Bergen, 1985). The input image was first filtered using two Gabor filters, in quadrature phase, *i.e.*, with a 90° phase relationship. These filters had a vertical preferred orientation, a preferred spatial frequency of 2 cycles per receptive field, and a spatial-frequency bandwidth of 1.6 octaves. The outputs of these filters was then squared and summed, generating the mean of a Poisson spike generator. The analytical expression of the complex-cell model is then

$$y(\mathbf{x}) = \langle \mathbf{f}_1, \mathbf{x} \rangle^2 + \langle \mathbf{f}_2, \mathbf{x} \rangle^2 \quad (2.18)$$

where \mathbf{f}_1 and \mathbf{f}_2 are the linear filters in quadrature phase. To vary the mean number of spikes in the responses we scaled the magnitude of the Gabor filters, generating responses with 5.0, 3.0, 1.0 and 0.5 spikes per image.

As before, to assess the quality of the estimated relevant space, we derived analytical relevant dimensions for the complex-cell model. We then compared them with their projection onto the estimated relevant space. Due to the linear filters in the first stage of the complex-cell model, the response to any image will equal, up to the noise, the response to the image projected in the span of the filters. Consequently, the two linear filters span the relevant space. Because these filters are orthogonal, after rescaling them to unit norm, they constitute a set of relevant dimensions.

Again, to judge the quality of the estimated kernels, we compared them with the true kernels. We derived them for the complex-cell model using the procedure described in Appendix 2.6.2. Only the second spatial derivatives with respect to \mathbf{x} of (2.18), evaluated at zero, are different from zero. Hence, from (2.39), only the second-order Volterra kernels of the complex cell are different from zero. Therefore, the complex cell can be represented with a second-order Volterra model.

2.3.2 Physiological Data

The physiological data used here were recorded in the laboratory of Dr. Yang Dan and the method used to obtain them were described by Touryan et al. (2005). Animal procedures to obtain these data were approved by the Animal Care and Use Committee at the University of California, Berkeley. Briefly, single-cell recordings were made in area 17 of adult cats (2-6.5 kg) using tungsten electrodes. Cells were classified as simple if the receptive fields had clear ON and OFF subregions and if the ratio of the first harmonic to the DC component of the responses to an optimally oriented drifting sinusoidal grating was greater than one. Natural images used in cell stimulation were selected at random from a database consisting of a variety of digitized movies (van Hateren & Ruderman, 1998), and the center patch (12×12 pixels) of each image was retained. Images were scaled so that they all had the same variance. These selected images had no temporal correlations and were presented at a frame rate of 24 Hz in an area slightly larger than the classical receptive field of the cell estimated by hand mapping. For analysis, Touryan et al. binned responses at 41.8 ms, the presentation frame rate, and the bin immediately following a frame was used as the response to that frame. They shared with us two to four repetitions of responses from four complex cells to 24,000 natural images. Below, we show the analysis of the cell for which we obtained the best predictions with both the Volterra and Touryan et al. models. For this cell, both models yielded better predictions when considering the spikes in the frame presentation bin as the cell response of to that

frame. We used the average response across four repetitions of the stimuli as the response variable to estimate the different models.

2.3.3 Data Partitioning

To control for possible overfitting of the different receptive field estimation models to the data used in their estimation, we partitioned the data into disjoint training and a test data sets. Only the training data set was used to estimate model parameters. The test data set was reserved to evaluate the models' predictive power. By using different training and test data sets, the prediction results reflected the generalization ability of the different models to predict responses to novel natural images.

In addition, between two to ten jackknifed data sets were generated by excluding different 10% segments of the training data set (Efron & Tibshirani, 1993). These jackknifed data sets were used to average out the noise from the estimated relevant dimensions and model parameters and to select model hyperparameters.

We used training sets of varying size (1,000, 3,000, 5,000, 10,000 and 15,000 image/response pairs). For the simulated and physiological data we used test sets of size 4,500 and 4,000 image/response pairs, respectively.

2.3.4 Receptive Field Estimation

2.3.4.1 Volterra Relevant-space Technique

Estimating the relevant dimensions: To develop low-dimensional Volterra models, one must estimate a set of relevant dimensions satisfying (2.8). Methods for doing this rely on different assumptions about the statistics of the input and/or about the properties of the response-generation function. The methods of reverse correlation (de Boer & Kuyper, 1968), spike triggered covariance (STC) (de Ruyter van Steveninck & Bialek, 1988) and maximally informative dimensions (Sharpee et al., 2004) have been introduced in the field of Neuroscience. These methods are powerful, but have constraints about the dimensionality of the relevant space, the type of stimulus used, or require an

arduous high-dimensional nonlinear optimization that constrains the dimensionality of estimable relevant spaces. (An exception may be the modification of STC for natural images explained in Section 2.3.4.4.) Other interesting methods involve perceptron neural networks (Hertz, Palmer, & Krogh, 1991) and partial least squares (Helland, 1988; Geladi & Kowalski, 1986). The former suffers from the curse of dimensionality and the latter relies on a linear assumption about the response-generation function. Other methods are sliced inverse regression and projection pursuit regression (PPR), both of which were introduced in Statistics. Sliced inverse regression (Li, 1991) makes no assumptions about the response-generation mechanism, but relies on the assumption of a spherical distribution of the stimuli. In turn, PPR (Friedman & Stuetzle, 1981) makes no assumption about the distribution of the inputs and uses the following generalized linear model for the response-generation mechanism

$$y(\mathbf{x}) = \bar{y} + \sum_{m=1}^{M_0} \beta_m \phi_m(\langle \mathbf{a}_m, \mathbf{x} \rangle) \quad (2.19)$$

with $E\{\phi_m(\langle \mathbf{a}_m, \mathbf{x} \rangle)\} = 0$ and $E\{\phi_m^2(\langle \mathbf{a}_m, \mathbf{x} \rangle)\} = 1$

Provided M_0 is sufficiently large, any smooth function can be represented by (2.19) (universal approximator property, Diaconis & Shahshahani, 1984). Another interesting property of PPR is that its estimation algorithm explicitly addresses the curse of dimensionality, by separately estimating each term in the generalized linear model. In the following experiments, we evaluated PPR for the estimation of the relevant space. In Appendix 2.6.3, we outline the PPR algorithm (see Friedman, 1984a, for further details). The estimated relevant dimensions for the low-dimensional Volterra model are computed by orthonormalizing the set of projection directions, $\{\mathbf{a}_1, \dots, \mathbf{a}_{M_0}\}$, in (2.19). To understand why this procedure works, let \mathcal{S} denote the space spanned by the projection directions, $\mathcal{S} = \langle \mathbf{a}_1, \dots, \mathbf{a}_{M_0} \rangle$. Let us also denote $\Pi_{\mathcal{S}}$ the projection operator onto the space \mathcal{S} . Because $\langle \mathbf{a}_i, \mathbf{x} \rangle = \langle \mathbf{a}_i, \Pi_{\mathcal{S}}(\mathbf{x}) \rangle$

$$y(\mathbf{x}) = \bar{y} + \sum_{m=1}^{M_0} \beta_m \phi_m(\langle \mathbf{a}_m, \mathbf{x} \rangle) = \bar{y} + \sum_{m=1}^{M_0} \beta_m \phi_m(\langle \mathbf{a}_m, \Pi_{\mathcal{S}}(\mathbf{x}) \rangle) = y(\Pi_{\mathcal{S}}(\mathbf{x})). \quad (2.20)$$

That $y(\mathbf{x}) = y(\Pi_{\mathcal{S}}(\mathbf{x}))$, implies that \mathcal{S} is a relevant space for y . Consequently, a set of relevant dimensions can be computed by orthonormalizing the set of projection directions $\{\mathbf{a}_1, \dots, \mathbf{a}_{M_0}\}$, as indicated above.

For the results reported here, we used the implementation of the PPR algorithm in R, an environment for statistical computing (Venables & Smith, 2002). This implementation selects initial values for the projection directions, \mathbf{a}_i in (2.19), by cross-correlating the input images with the residuals of the responses. Consequently, it might fail to estimate the relevant dimensions of cells for which the responses are completely uncorrelated with the input images. For these cases one can use the implementation of PPR developed by Roosen (1994), which allows the specification of initial projection directions. However, for all the conditions reported below, there was enough correlation between images and responses for the R implementation of PPR to estimate good relevant dimensions.

The estimation of the projection directions requires two runs of the PPR algorithm. The first run estimates the number, L , of projection directions needed to characterize the cell. Then, the second run estimates the L projection directions. In the first run, for each jackknifed data set in our training set (Section 2.3.3), we ran the PPR algorithm with parameters $M_0 = 1$ and $M = 6$. These parameters allowed assessing the predictive error of PPR models with one to six projection directions (see Appendix 2.6.3). For each jackknifed data set, PPR generates a predictive-error-versus-number-of-projection-directions curve. The PPR predictive-error curves of different jackknifed data sets were averaged. This average generated a new curve that was used to select the number of projection directions. PPR predictive-error-versus-number-of-projection-directions curves follow a common pattern. Before a critical number of terms, the predictive error decreases substantially as we increase the number of terms. Increasing the number of terms beyond the critical value yields only small reductions in predictive error. We selected the number of

projection directions as the largest number, between one and six, for which the error bars (size ten standard errors) did not intersect those of the previous number of projection directions.

After choosing the number of PPR projection directions, L , we proceeded to estimate the projection directions. We ran the PPR algorithm, with parameters $M_0 = L$ and $M = 6$, for the N ($2 \leq N \leq 10$) jackknifed data sets, estimating N sets of L noisy projection directions. To attenuate the effect of noise, we averaged these sets of projection directions, as described in Section 2.3.5.

Note that PPR uses raw natural images as inputs, in contrast to spike-triggered techniques, which use spike-triggered images, *i.e.*, images weighted in proportion to their corresponding response.

Estimating the low-dimensional Volterra model: Having estimated the relevant dimensions, we constructed the low-dimensional Volterra model (2.7). We estimated its parameters, q_i , by minimizing the mean-square error between the responses of the cell model and the predictions of the low-dimensional Volterra model. As noted in Section 2.2.2, the solution to (2.13) is a linear problem. We solved it using the pseudoinverse computed from the singular value decomposition (Mendel, 1995).

Selecting the order of Volterra models: For each training jackknife data set (Section 2.3.3), we estimated low-dimensional Volterra models of different orders. We then evaluated the predictive power of these models using the 10% portion of the training data not included in the jackknife data set. The predictive power was the cross correlation between cell responses and Volterra model predictions. We then generated a curve of predictive power versus model order. Curves of different data sets were averaged. The least order maximizing the average predictive power was selected as the order of the Volterra model.

2.3.4.2 Spike-triggered Average

Suppose the responses of a cell are generated by a static nonlinearity on the projection of input images along a single relevant dimension. In other words, suppose $y(\mathbf{x}) = N(\langle \mathbf{h}, \mathbf{x} \rangle) + \epsilon$, where $y(\mathbf{x})$ is the response of the cell to image \mathbf{x} , \mathbf{h} is a linear filter, $\langle \cdot, \cdot \rangle$ is the inner-product operation and N is a static nonlinearity. When the images are Gaussian white noise, the filter \mathbf{h} is proportional to the cross-correlation between the inputs and the responses (P. Marmarelis & Marmarelis, 1978).

However, natural images are not white noise. For them, when the cell can be represented as a linear device, the cross-correlation between the stimuli and the responses equals the product of the autocorrelation of the inputs and the filter, *i.e.*, $\mathbf{C}_{yx} = \mathbf{C}_{xx}\mathbf{h}$. Then, for non-white inputs and a linear cell, the filter \mathbf{h} can be recovered as $\mathbf{h} = \mathbf{C}_{xx}^{-1}\mathbf{C}_{yx}$. The autocorrelation matrix, \mathbf{C}_{xx} , for natural stimuli is nearly singular. Therefore, its true inverse tends to amplify noise. To avoid this problem, we regularized the autocorrelation matrix using the truncated singular value decomposition (Hansen, 1987) and computed the pseudoinverse (Ben-Israel & Greville, 1980) from this regularized matrix. The computation of the truncated singular value decomposition uses a threshold to decide how many singular values to include in the regularized matrix. For each condition in the studies reported below, we selected the optimal threshold by using k-fold cross-validation (Efron & Tibshirani, 1993). We used the jackknifed data sets from the training data (Section 2.3.3) to estimate different STA filters. To minimize the effect of noise, the average of these filters was used as the final STA estimate.

2.3.4.3 Histogram-based Model for Simple Cells

The histogram-based model for simple cells characterizes the response to an image \mathbf{x} as a linear filter, \mathbf{h} , followed by a static one-dimensional nonlinearity (Chichilnisky, 2001). Precisely, $y(\mathbf{x}) = N(\langle \mathbf{h}, \mathbf{x} \rangle) + \epsilon$. We estimated the linear filter using STA and the static nonlinearity by fitting the parameters of the same sigmoidal function used to generate the simulated data to a histogram of responses as a function of projections of input images

in the estimated filter. We used histograms with equi-spaced bins and selected the bin size from the data range using the Sturges rule (Sturges, 1926).

2.3.4.4 Spike-triggered Covariance for Natural Images

For Gaussian white-noise stimuli, consider stimuli that elicit spikes (spike-triggering stimuli). The variance of the projections of these stimuli along the cell’s relevant dimensions should be larger or lower than the variance along non-relevant dimensions. The dimensions with large or low variance correspond to the eigenvectors of the autocovariance matrix with large or low eigenvalues respectively. For Gaussian stimuli, Spike-Triggered Covariance (STC) (de Ruyter van Steveninck & Bialek, 1988; Simoncelli et al., 2004) identifies these “extreme” eigenvectors as the relevant dimensions of a cell.

Touryan et al. (2005) have proposed a modification of STC for natural stimuli. This modification starts by whitening the natural stimuli. Denote by U the matrix of eigenvectors of the autocovariance of the stimuli (one eigenvector per column), and by λ_i its eigenvalues. The matrix of normalized eigenvectors is defined as

$$U_n = U \begin{pmatrix} \frac{1}{\sqrt{\lambda_1}} & & 0 \\ & \ddots & \\ 0 & & \frac{1}{\sqrt{\lambda_n}} \end{pmatrix} \quad (2.21)$$

Then the whitened natural images are $X_w = XU_n$. STC for natural images now performs a classical STC analysis on the whitened natural images, estimating a set of relevant dimensions, V_w (one relevant dimension per column). Finally, the estimated relevant dimensions of the cell, V (one relevant dimension per column), are $V = U_n V_w$.

The autocovariance matrix of natural images is nearly singular, so its last eigenvalues ($\lambda_i, i \gg 1$) will be very small and tend to amplify noise. To avoid this effect, a threshold is selected and eigenvalues below this threshold are set to one. In this way, only the eigenvectors corresponding to large eigenvalues are normalized in (2.21). With

physiological and simulated data, we used different threshold values for different cells and selected the value maximizing the predictive power. Best predictions were obtained using thresholds that normalized around 35% of the eigenvectors. So, for the results reported below, we used this criterion to select the threshold. The same criterion was used by Touryan et al. (2005). Multiple sets of STC relevant dimensions were estimated using the jackknifed data sets from the training data (Section 2.3.3). We then averaged these sets (Section 2.3.5) to obtain the final STC estimates.

2.3.4.5 Histogram-based model for Complex Cells

To reconstruct the nonlinear function from the relevant space by a histogram method, we use the technique proposed by Touryan et al. (2005). We construct two independent histograms for responses as a function of projections of images in each of the two estimated relevant dimensions. We then fit a power function, $f(p) = \beta|p|^\gamma + r_0$ to each histogram (where $f(p)$ is the response of the cell to a given image, p the projection of the image onto a estimated relevant dimension, and β , γ , and r_0 are free parameters). The reconstructed nonlinearity is then the sum of the power functions fitted to each histogram, $y(\mathbf{x}) = f_1(\langle \phi_1, \mathbf{x} \rangle) + f_2(\langle \phi_2, \mathbf{x} \rangle)$ (where $y(\mathbf{x})$ is the response of the cell to image \mathbf{x} , f_1 and f_2 are the power functions fitted to the histograms, ϕ_1 and ϕ_2 are the first and second relevant dimensions, and $\langle ., . \rangle$ is the inner product operation).

2.3.5 Averages of Sets of Relevant Dimensions

We wish to average N sets of L estimated relevant dimensions, when each set has been estimated from a different portion of the training data. All sets describe, with noise, a common relevant space. The goal is to estimate this common space. Because different sets might represent the common space in different coordinate systems, we cannot average them element by element. To perform the average, we instead collected the N sets into a matrix A . In A , columns $\{(i-1)L, \dots, iL\}$ correspond to the relevant dimensions of the i^{th} set. The column space of A spans an L -dimensional relevant space (signal) and a noise

space. We use the singular-value decomposition (SVD) to separate the column space of A into a signal and an orthogonal noise space (Scharf, 1991). Specifically, we compute the SVD of A , $A = U\Sigma U^T$, and use the first L left singular vectors (first L columns of matrix U) to represent the signal space. Thus, the first L columns of matrix U are the average of the N sets of L relevant dimensions.

2.3.6 Scaling Kernels

To display kernels and to measure the goodness of fit of estimated kernels, we scaled them in proportion to their mean contribution to the response of the cell. The first-order kernel was scaled by the mean absolute value of the stimuli. In turn, the second-order kernel was scaled by the mean absolute value of the autocorrelation. Precisely, denoting the mean of z by \bar{z} , the scaled first- and second-order kernels are $k_1^s(i, j) = k_1(i, j) \overline{|x(i, j)|}$ and $k_2^s(i_1, j_1, i_2, j_2) = k_2(i_1, j_1, i_2, j_2) \overline{|x(i_1, j_1)x(i_2, j_2)|}$, respectively.

2.3.7 Displaying Kernels

In this paper, we represent first- and second-order spatial Volterra kernels slices in three-dimensional (3D) spaces. We display 3D spaces as perspective and contour plots. First-order kernels $k_1(i, j)$, have two independent variables, the dimensions of space, and one dependent variable, the amplitude. The latter is represented on the vertical axis of the 3D plots, with i and j along the X and Y axes respectively. Second-order kernels, $k_2(i_1, j_1, i_2, j_2)$, have four independent variables and one dependent variable. We display second-order kernel slices respect to different reference positions. That is, we fix the reference position (i_1, j_1) and plot the value $k_2(i_1, j_1, i_2, j_2)$ on the vertical axis, with i_2 and j_2 on the X and Y axes respectively. These plots show contributions to the response of interactions between individual pixels of the image and pixel $x(i_1, j_1)$. The kernels shown in Section 2.4 have been scaled (Section 2.3.6). This allows the reader to appreciate visually the average contribution of a kernel to the cell response and to

compare the contributions from different kernels. For predicting responses, on average, larger kernels are more relevant than smaller ones.

2.3.8 Goodness of Fit of Kernels

To measure the goodness of fit of estimated to true kernels we scaled them (Section 2.3.6), and computed their mean-square error (MSE). The MSE is a scale sensitive measure. This implies that if true and estimated kernels have small amplitude, then the MSE between them will be small, independently of how well they correlate. For measuring errors between scaled kernels, this scale sensitivity is a good property. This is so because scaled kernels with small amplitudes contribute little to the response and thus it does not matter how well the estimated correlates with the true kernel. Moreover, the MSE allows the comparison of errors across different kernels. This happens because the amplitude of scaled kernels indicates their mean contribution to the response. Therefore, a similar MSE for different scaled kernels indicates that they contribute similar errors to the response.

2.3.9 Volterra terms contributions

To assess the importance of individual Volterra terms to the response, we decomposed (2.3) as a sum of terms of increasing nonlinear order,

$$y(\mathbf{x}) = y_0 + y_1(\mathbf{x}) + \dots + y_Q(\mathbf{x}), \quad (2.22)$$

where

$$y_0(\mathbf{x}) = k_0, \tag{2.23}$$

$$y_1(\mathbf{x}) = \sum_{i,j=1}^N k_1(i,j)x(i,j), \tag{2.24}$$

...

$$y_Q(\mathbf{x}) = \sum_{i_1,j_1,\dots,i_Q,j_Q}^N k_Q(i_1,j_1,\dots,i_Q,j_Q)x(i_1,j_1)\dots x(i_Q,j_Q) \tag{2.25}$$

We wanted to know how relevant was each term relative to the others for the generation of the response. For this purpose, we defined the contribution of the i th term to the response $y(\mathbf{x})$ to image \mathbf{x} as $contrib_i(\mathbf{x}) = |y_i(\mathbf{x})| / \sum_{j=0}^Q |y_j(\mathbf{x})|$.

The definition of contribution is such that if for a given image the contribution of a specific term is large, then, if we set to zero the component of the response due to that term, the response is largely modified. On the other side, if the contribution of a term is small, then the response is mostly unaffected when we set to zero the component of the response due to that term.

2.4 Results

We initially evaluated the Volterra relevant-space technique with models of cortical simple and complex cells. The use of simulated data allowed us to derive analytically true relevant dimensions and the true Volterra kernels of the cell models. We could then compare these true parameters with those estimated from input/output data. The outcome of these comparisons appear in Sections 2.4.1.1 and 2.4.2.1. Next, we compared the Volterra model with two other well-known models that also use relevant dimensions. For the simulated simple cell, the comparison was with a model using STA to recover a single relevant dimension and histograms to estimate the nonlinearity (Chichilnisky, 2001). For the simulated complex cell, the comparison was with a model using STC to recover the relevant space and histograms to estimate the nonlinearity (Touryan et al., 2005). In Sections 2.4.1.2 and 2.4.2.2, we present the results of comparing PPR with STA and STC respectively. In turn, Sections 2.4.1.3 and 2.4.2.3, compare the Volterra model with the histogram-based models for, respectively, the simulated simple and complex cells. Finally, we use physiological data from cortical complex cells to show the feasibility of our methods with real data (Section 2.4.3).

2.4.1 Simulated Simple Cell

2.4.1.1 True Parameters

We used Projection Pursuit regression (PPR) to estimate the relevant space of the simulated simple cell. For this estimation, we used a training set of 5,000 patches from natural images. Fig. 2.2a shows the curve of predictive error versus the number of estimated relevant dimensions for the simple cell. Using two estimated relevant dimensions leads to a large reduction in prediction error compared to using only one dimension. In contrast, using more than two estimated relevant dimensions yield only small reductions in predictive error. (Here, we define large as the lack of intersection between neighboring error bars – size ten standard errors.) Based on these data, we used the projection directions

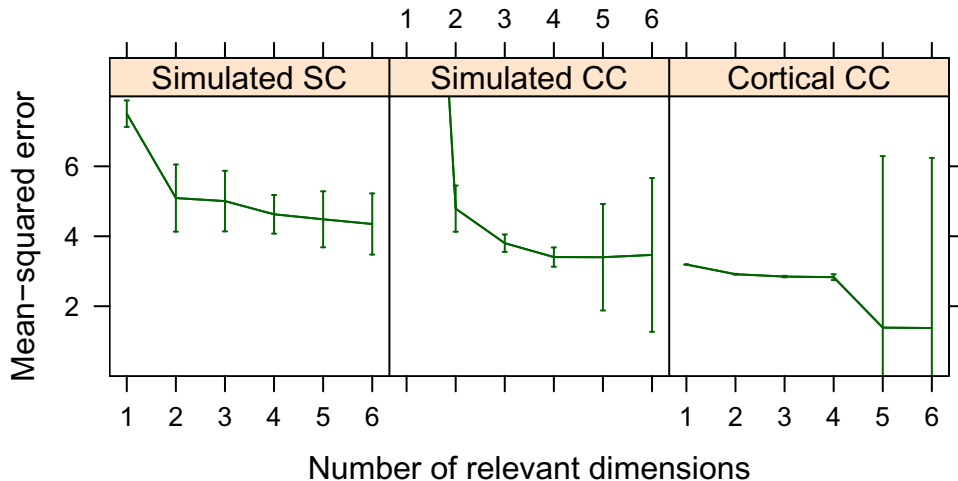


Figure 2.2: Goodness of fit of PPR models as a function of number of estimated relevant dimensions. (a) Simulated simple cell. (b) Simulated complex cell. (c) Cortical complex cell. The size of error bars is 10 standard errors. Based on these data, we use two, two, and three estimated relevant dimensions for the Volterra model of the simulated simple cell, simulated complex cell, and cortical complex cell respectively.

of a two-term PPR model to compute the estimated relevant dimensions for the Volterra model (Section 2.3.4.1).

Although we estimated two significant relevant dimensions from the data, as described in Section 2.3.1.1, the relevant space of the simple-cell model is spanned by a single relevant dimension. What matters for our method is that this space is part of the larger, estimated one. If this holds, then the estimation of the Volterra model, by minimization of (2.13), can prune non-relevant estimated space. (This pruning could be done, *e.g.*, by setting appropriate coefficients to zero.) We must thus ensure that the true relevant dimension is similar to its projection onto the estimated relevant space. The left column of Fig. 2.3 shows the analytical relevant dimension of the simple-cell model. The right column shows the projection of this dimension onto the estimated relevant space. Despite the effects of noise, the projection of the true relevant dimension onto the estimated

relevant space captures well the preferred orientation, spatial frequency, and phase of the analytical one.

To measure the goodness of fit of the true relevant dimension and its projection onto the estimated relevant space, we plot the coefficients of the true relevant dimension against the coefficients of its projection. We then fit a linear model to this plot and use the coefficient of determination (or r^2 statistic) of the fitted linear model as the goodness-of-fit measure. The r^2 statistic is the ratio between the reduction in variance achieved by the regression model (the total variance of the outputs minus the variance of the residuals) and the total variance of the outputs. Without a relationship between the true relevant dimension and its projection onto the estimated relevant space, the r^2 statistic would be zero. In contrast, if the true relevant dimension were identical to its projection, then the r^2 statistic would be 1. For the simple cell, the r^2 statistic for the analytical relevant dimension and its projection onto the estimated relevant space was 0.94.

Having estimated the relevant dimensions, we next estimated Volterra models of different orders using the same set of 5,000 image/response pairs used to estimate the relevant dimensions. We selected the order of the Volterra model as the least order maximizing the average predictive power (Section 2.3.4.1). Fig. 2.4a shows the mean predictive power of simple-cell Volterra models as a function of their order. Second- and third-order Volterra models lead to large improvements in cross-correlation (Here, large is defined as the lack of intersection between neighboring error bars – size two standard errors), illustrating the relevance of nonlinear contributions to characterize the simple-cell data. Volterra models of order higher than three do not lead to improvements in predictive power. Based on these data, we selected a third-order Volterra model.

How important are high-order nonlinearities in the description of responses? To answer this question we defined a simple measure of the relative importance of each Volterra term to the overall response (Section 2.3.9). We expected that the relative contribution to the response of different Volterra terms should be dependent on the response range. Higher responses might be more nonlinear and thus, would require higher-order Volterra

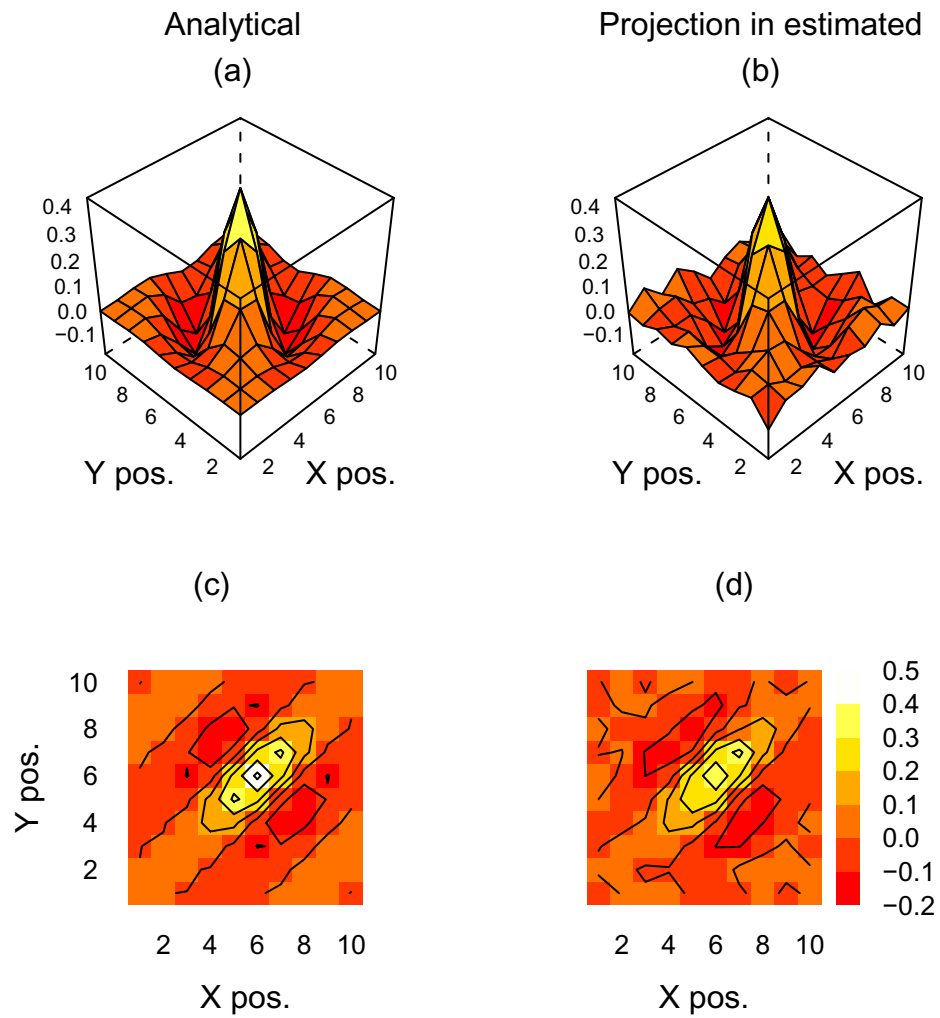


Figure 2.3: Simulated simple cell: relevant dimensions. (a), (c): analytical relevant dimension. (b), (d): projection of the analytical relevant dimension onto the estimated relevant space. (a), (b): perspective plot. (c), (d): contour plot. The analytical relevant dimension is accurately approximated by its projection onto the relevant space, $r^2 = 0.94$.

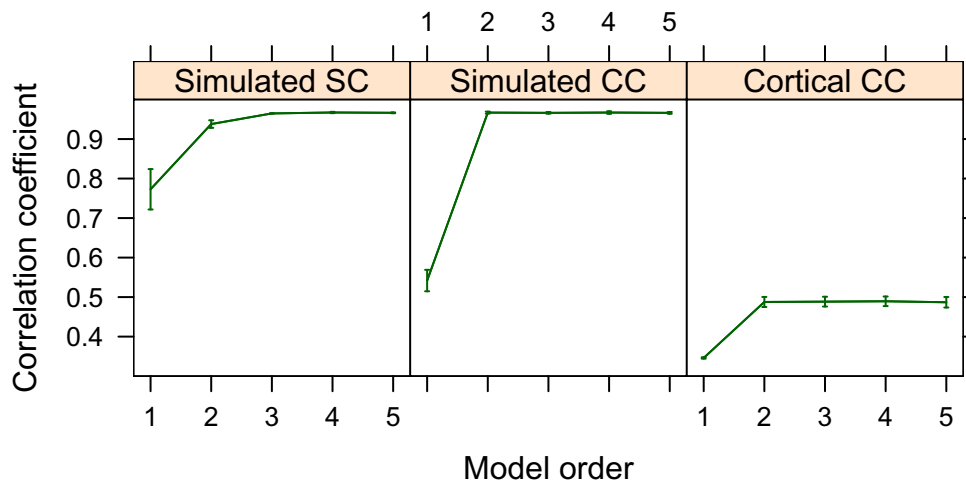


Figure 2.4: Correlation coefficients between cell responses and the Volterra-model predictions versus the order of the Volterra model. (a) Simulated simple cell. (b) Simulated complex cell. (c) Cortical complex cell. The size of error bars is two standard errors. The order of the Volterra model was selected as the least order maximizing the average predictive power. We selected a 3rd, 2nd, and 2nd order for the simulated simple cell, simulated complex cell, and cortical complex cell respectively.

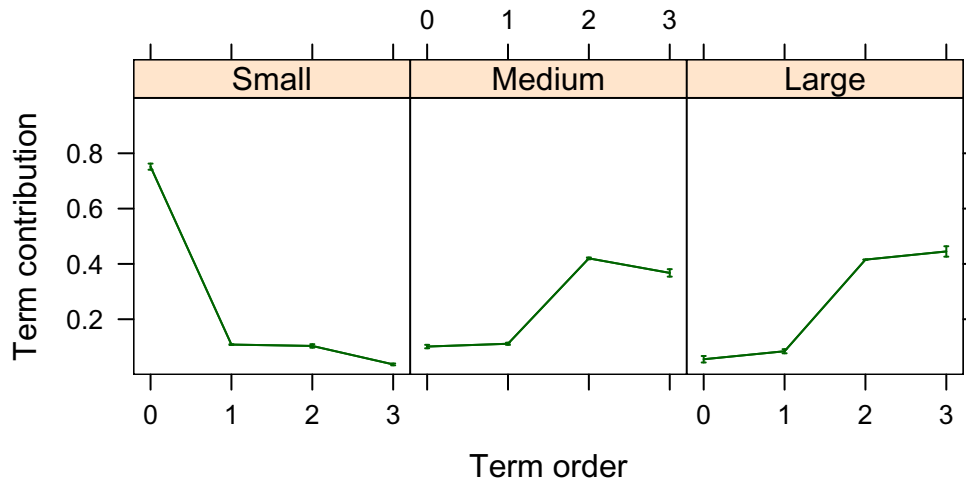


Figure 2.5: Simulated simple cell: mean relative contributions of Volterra terms to the response. Contributions have been averaged over different response ranges: (a) small responses (first quartile), (b) medium responses (second and third quartiles), and (c) large responses (fourth quartile). As the responses increase, high-order Volterra terms become more important.

terms. Fig. 2.5 shows the mean relative contributions of different Volterra terms to the response of the simple cell. We show the mean contributions for groups of small (first quartile), middle (second and third quartiles), and large (fourth quartile) responses. For small responses (Fig. 2.5a), the most important terms are the low-order ones. In contrast, for medium (Fig. 2.5b) and large (Fig. 2.5c) responses, the situation reverses and high-order terms become dominant. Therefore, even for a cell often considered linear as the simple cell, high-order nonlinearities may matter, specially when describing responses to stimuli causing large responses. In the natural world, most responses may be small, as contrasts tend to be low (Balboa & Grzywacz, 2000; Tadmor & Tolhurst, 2000; Ruderman & Bialek, 1994; Zhu & Mumford, 1997), but some responses will be high, as the distribution of natural intensities is kurtotic (Field, 1987).

To assess the quality of the estimated kernels, we compared them with the true kernels derived from the analytical expression of the simple-cell model (Appendix 2.6.2). The

left-column in Fig. 2.6 shows the first-order true kernel, whereas its estimation appears in the right-column. Similarly, Fig. 2.7 shows the second-order kernel slice with respect to Position (6, 6), this was the slice with largest error. The kernels have been scaled to reflect their mean contribution to the cell response (Section 2.3.7).

To measure the goodness of fit of estimated to true kernels we scaled them, in proportion to their mean contribution to the response, and computed their MSE (Section 2.3.8). For the simple cell, the MSE between scaled true and estimated kernels was 2.42E-03 for the first-order kernel (Fig. 2.6) and 1.83E-03 for the second-order kernel slice with respect to Position (6, 6) (Fig. 2.7). This slice had the largest MSE among all second order kernels slices (the MSE averaged across all second-order kernel slices was 1.74E-04). Therefore, despite the noise, the estimated kernels accurately approximate their true values.

Although for the simple cell the first-order kernel is similar to the second-order kernel slice with respect to Position(6, 6), their interpretation is different. The first-order kernel indicates that the natural image would elicit response if it were bright in the yellow region and dark in the red region. In turn, the second-order kernel slice indicates that simultaneous bright stimuli anywhere in the yellow region and at Position (6, 6) would facilitate the response. However, having bright stimuli in the red regions and at Position (6, 6) at the same time would inhibit the response. In other words, first-order kernels describe how much a stimulus at a given position linearly contributes to the response. Second-order kernels describe nonlinear modulations of responses, through interactions between stimulus pairs at given positions.

2.4.1.2 PPR versus STA

So far, we showed that PPR is a good technique to estimate relevant dimensions to use in the construction of Volterra models of simulated simple cells. Here, we compare PPR with STA, a technique often used to estimate the relevant dimension of simple cells. In particular, we compared the convergence properties of both algorithms parametric on number of inputs and signal-to-noise ratio. Fig. 2.8a plots the r^2 statistic for the regression

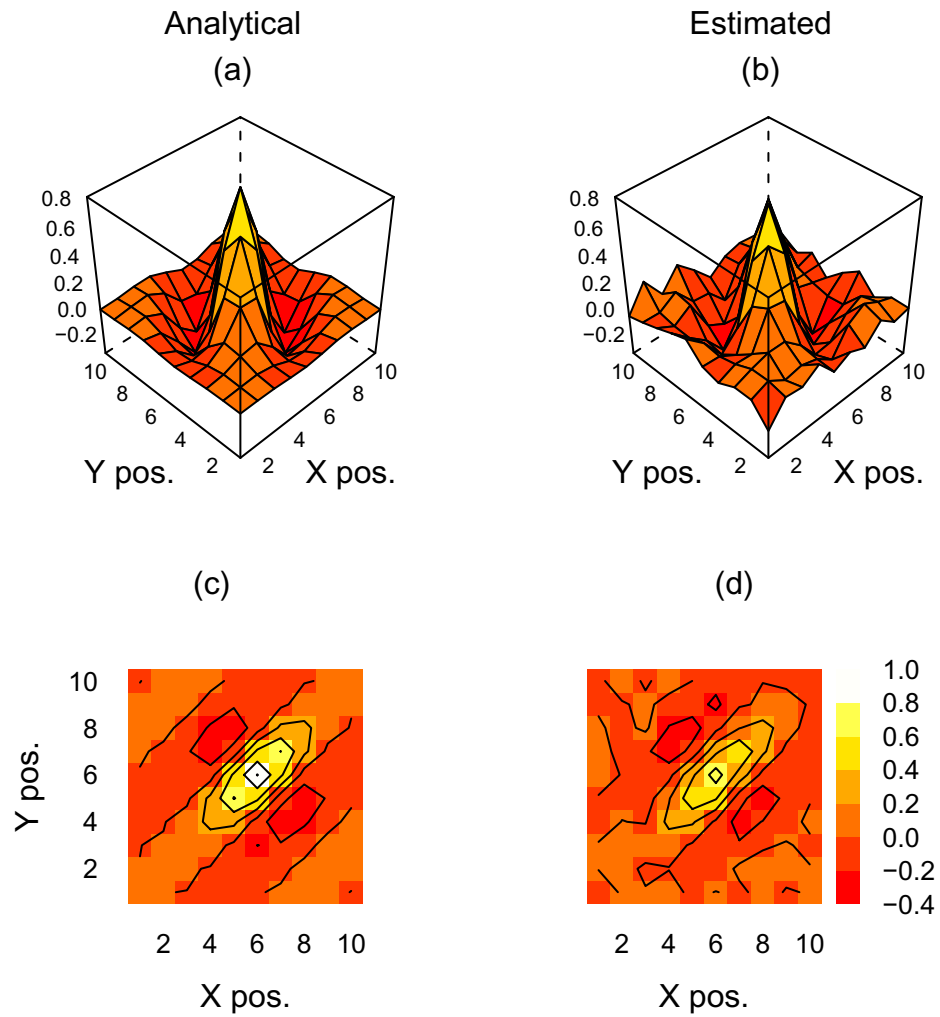


Figure 2.6: Simulated simple cell: first-order kernels. (a), (c): true. (b), (d) estimated. (a), (b): perspective plot. (c), (d): contour plot. Despite the noise, the estimated kernel accurately approximates its true value, $MSE=2.42E-03$.

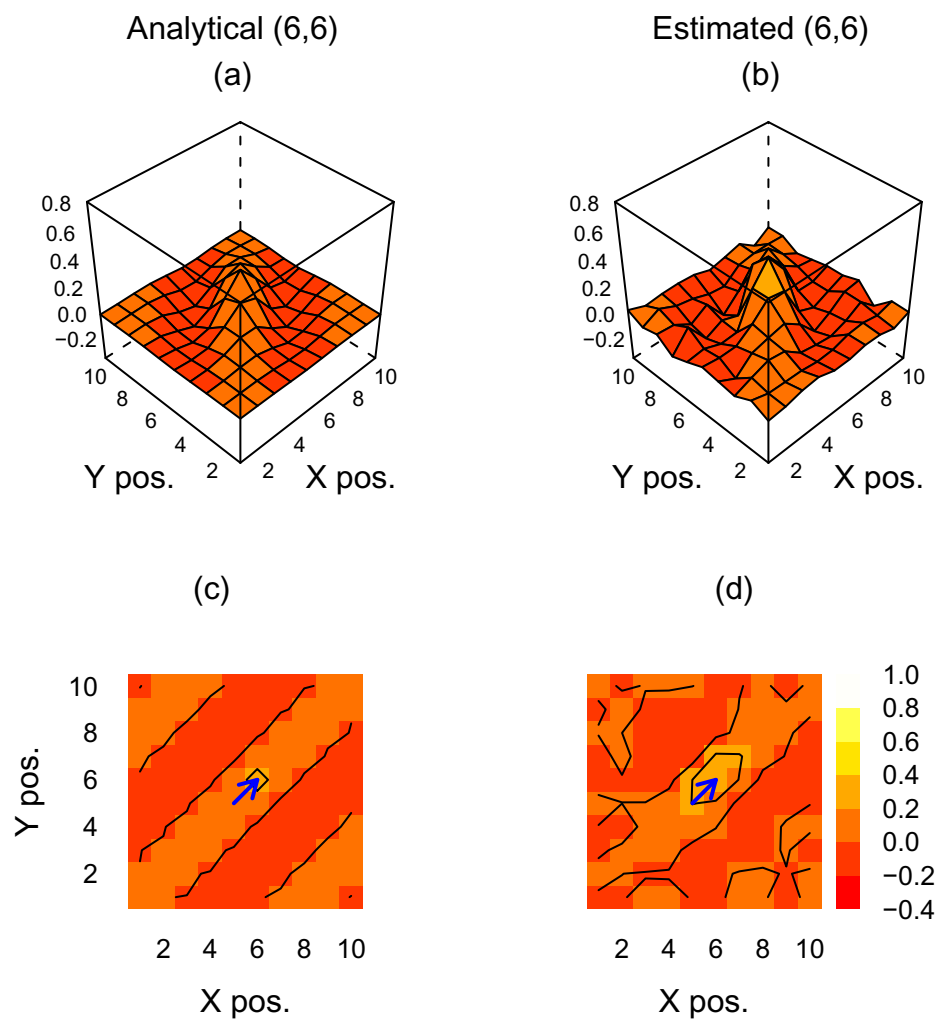


Figure 2.7: Simulated simple cell: second-order kernel slices with respect to Position (6, 6). (a), (c): true. (b), (d) estimated. (a), (b): perspective plot. (c), (d): contour plot. Blue arrows in the contour plots indicate the reference position. Despite the noise, the estimated kernel accurately approximates its true value, $MSE=1.83E-03$.

between the coefficients of the true and estimated relevant dimensions as a function of the number of inputs used in the estimation. For this plot, the average response of the simulated simple cell was 5 spikes/image. Fig. 2.8b plots the same r^2 statistic as a function of the number of spikes per image in the simulated responses, when using 5,000 inputs in the estimation. These figures show that the two PPR relevant dimensions better approximate the true relevant dimension of the simple-cell model than the relevant dimension estimated by STA (black versus green curve). The superiority of PPR over STA is not due to the fact that PPR is using two relevant dimensions to approximate the true relevant dimension, while STA is using only one. PPR also outperforms STA when using one relevant dimension (red versus green curves).

2.4.1.3 Volterra model versus histogram-based model

Section 2.4.1.1 showed that Volterra models can represent the nonlinearities of simple cells well, starting from its estimated relevant dimension. We now compare the Volterra model with another popular technique to represent nonlinearities, namely, a histogram-based model (Section 2.3.4.3). To make our comparison independent from the method used to estimate the relevant space, we compare the predictive power of the Volterra model using the two PPR relevant dimensions, the Volterra model using the STA relevant dimension, the nonlinearity estimated from histograms using the first PPR relevant dimension, and the nonlinearity estimated from histograms using the STA relevant dimension. (We use only the first PPR relevant dimension with the histogram-based model, since it is one-dimensional. In other words, this model accepts only one relevant dimension as input.) The predictive power of these models as a function of the number of inputs (using a mean response of 5 spikes/image) is plotted in Fig. 2.9a. A plot as a function of the number of spikes/image (using 5,000 images) appears in Fig. 2.9b. Note that, as discussed in Sec. 2.3.3, the data used to evaluate the different models was different from that used to estimate their parameters. Thus, the predictive power measures are not inflated by over-fitting effects.

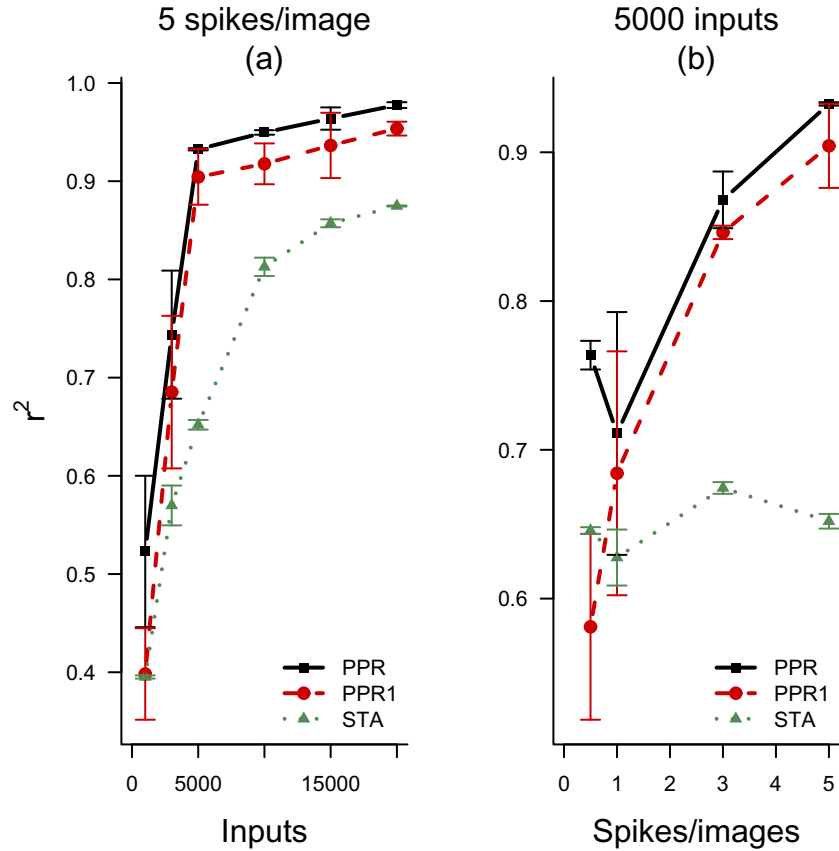


Figure 2.8: Simulated simple cell: comparison between PPR and STA. The figures show the r^2 statistic for the regression of the coefficients of the true and estimated relevant dimensions. This statistic appears as a function of (a) the number of inputs, when the mean response of the simulated cell is 5 spikes/image, and (b) the mean number of spikes in the responses, when using 5,000 image/response pairs. Black curve: r^2 statistic between the true relevant dimension and its projection onto a space spanned by two PPR relevant dimensions. Red curve: r^2 statistic between the true and one PPR relevant dimensions. Green curve: r^2 statistic between the true and STA relevant dimensions. The size of the error bars is two standard errors. PPR performs better than STA either when using one (red versus green curve) or two (black versus green) relevant dimensions to approximate the true relevant dimension.

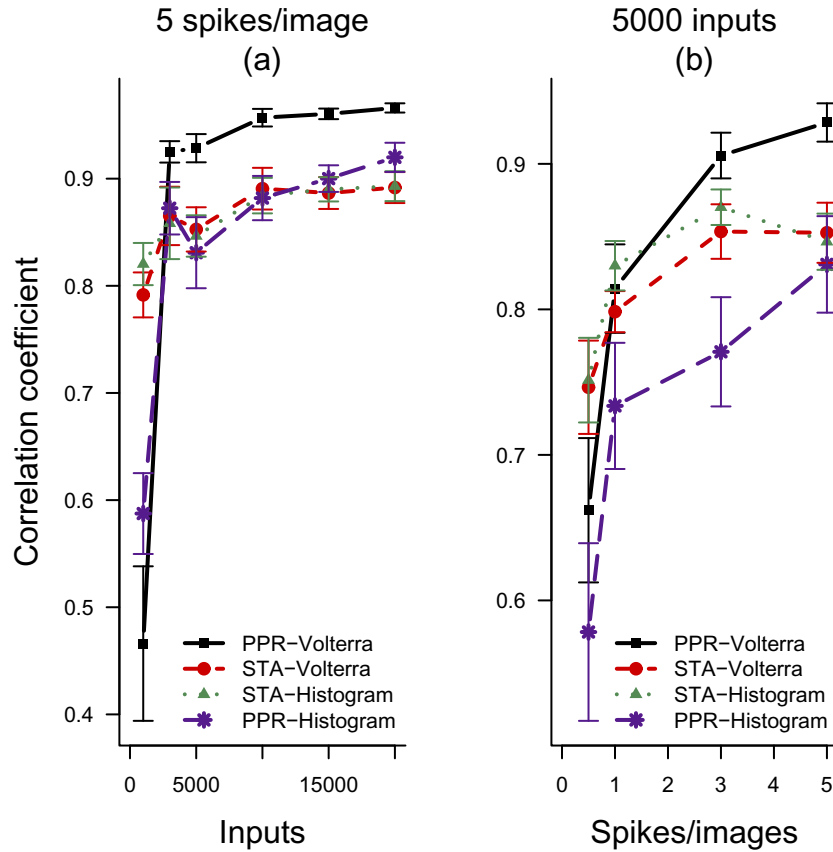


Figure 2.9: Simulated simple cell: predictions of the Volterra model and the nonlinearity estimated from histograms. This figure shows average correlation coefficients between simulated cell responses and model predictions. These coefficients appear as functions of (a) the number of inputs (using a mean response of 5 spikes/image) and (b) the number of spikes per image (using 5,000 image/response pairs). Black curve: Volterra model using the two PPR relevant dimensions. Red curve: Volterra model using the STA relevant dimension. Green curve: Nonlinearity estimated from histograms using the STA relevant dimension. Blue curve: Nonlinearity estimated from histograms using the first PPR relevant dimension. The size of error bars is two standard errors. For sufficiently high number of inputs and signal-to-noise ratio, the Volterra model using the relevant dimensions estimated by PPR performs better than the nonlinearity estimated from histograms. When using the STA relevant dimension, both models perform equally.

The figures show that, when using the relevant dimensions estimated by PPR, the Volterra model predicts better than the nonlinearity estimated from histograms. This superiority holds for moderate to high number of inputs, *i.e.*, more than 1,000 image/response pairs, and for moderate to high signal-to-noise ratios, *i.e.*, more than 1 spike per image. When using the STA relevant dimension, the Volterra model performs at the same level as the nonlinear function estimated from histograms. This is significant, because the simulated data accurately matches the assumptions of the STA-histogram model. Moreover, the parametric nonlinear function fitted to the histogram in the reconstruction of the nonlinear function (Section 2.3.4.3) was the same sigmoidal function used to generate the simulated responses (Section 2.3.1.1). Therefore, that the Volterra model is at least as good, making no assumptions on the response-generation mechanism is important.

2.4.2 Simulated Complex Cell

2.4.2.1 True Parameters

As for simple cells, we also used PPR to estimate the relevant space of the simulated complex cell from 5,000 stimulus/response pairs. Figure 2.2b shows the mean error as a function of the estimated number of relevant dimensions. Two is the largest number of estimated relevant dimensions for which models predict significantly better than models using fewer dimensions. Consequently, we computed the relevant dimensions for the Volterra model from a two-term PPR model.

The left columns of Fig. 2.10 and Fig. 2.11 show the first and second analytical relevant dimensions of the complex-cell model. In turn, the right columns show their projection onto the space spanned by the two estimated relevant dimensions. The analytical relevant dimensions are accurately approximated by their projection onto the estimated relevant space (first relevant dimension $r^2 = 0.96$, second relevant dimension $r^2 = 0.95$).

With these relevant dimensions we estimated Volterra models of different orders, using the same set of 5,000 stimulus/response pairs as for the estimation of the relevant

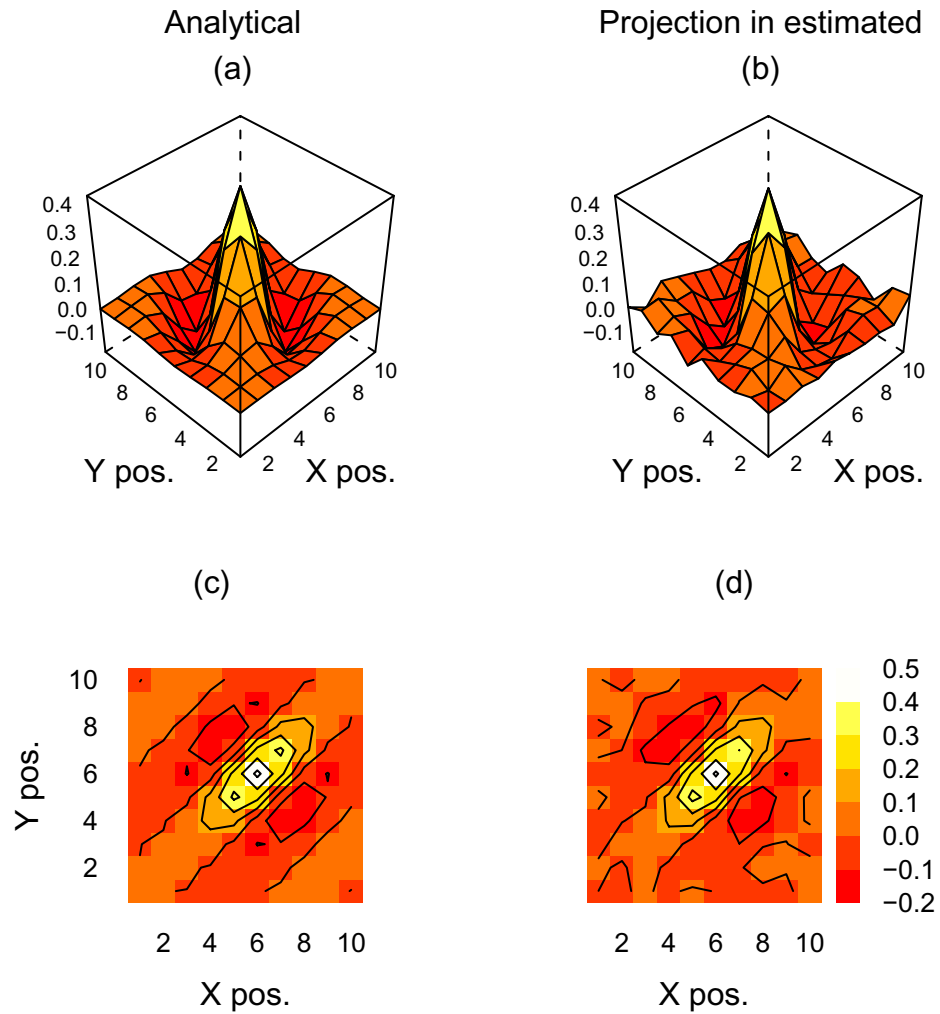


Figure 2.10: Simulated complex cell: first relevant dimension. (a), (c): first analytical relevant dimension. (b), (d): projection of first analytical relevant dimension onto the estimated relevant space. (a), (b): perspective plot. (c), (d): contour plot. The analytical relevant dimension is well approximated by its projection onto the estimated relevant space, $r^2 = 0.96$.

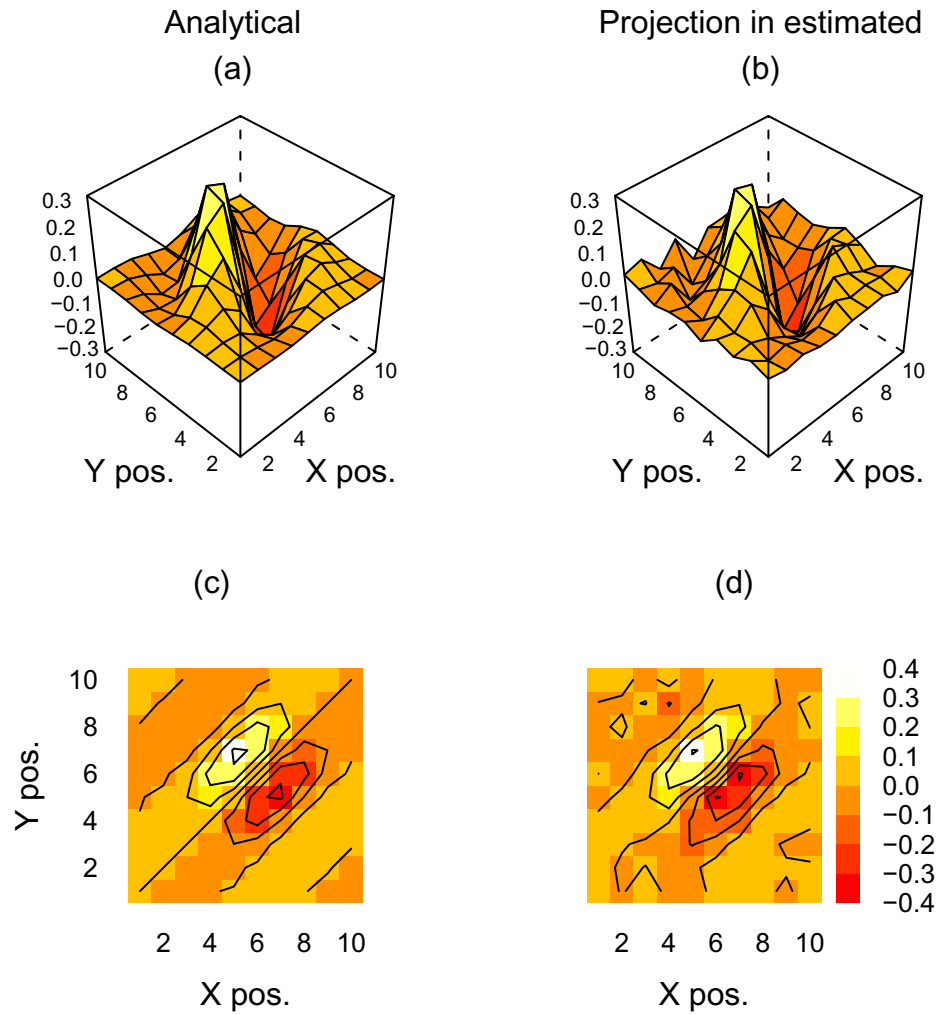


Figure 2.11: Simulated complex cell: second relevant dimension. (a), (c): second analytical relevant dimension. (b), (d): projection of second analytical relevant dimension onto the estimated relevant space. (a), (b): perspective plot. (c), (d): contour plot. The analytical relevant dimension is well approximated by its projection onto the estimated relevant space, $r^2 = 0.95$.

dimensions. We selected the order of the Volterra model as the least order maximizing the average predictive power (Section 2.3.4.1). Fig. 2.4b shows the mean predictive power of complex-cell Volterra models as a function of their order. Linear models predict the responses poorly due to a model mismatch. A linear model cannot accurately predict a quadratic response. Second- and higher-order Volterra models perform equally well. So, we used a second-order Volterra model to characterize the complex-cell data.

As for the simulated simple cell, we looked at the contributions of the various Volterra terms to the response. We again averaged the contributions across small (first quartile), medium (second and third quartiles), and large (fourth quartile) responses. These contributions are shown in Fig. 2.12. The simulated complex cell is purely second order, so the second-order Volterra term should be the only one that contributes to the response. Accordingly, medium and large responses are dominated by the second-order term. However, the spontaneous, zeroth-order, Volterra term has been spuriously fitted to a non-zero value. Its contribution is largest for small responses. Such spurious fits may occur because, at small responses, the Poisson distribution is highly kurtotic, admitting values that would be outliers in Gaussian distributions. Consequently, our least-square fit, that is sensitive to outliers, might not be sufficiently robust at small responses. Nevertheless, these plots indicate that one should focus on the second-order kernel to interpret the responses of the simulated complex cell.

To assess the quality of the estimated kernels for the complex cell, we compared them with their true values. The left column in Fig. 2.13 shows the first-order true kernel, whereas its estimation appears in the right column. Similarly, Fig. 2.14 shows the second-order Volterra kernel slice with respect to Positions (6, 6). The kernels have been rescaled to reflect their mean contributions to the cell response (Section 2.3.7).

To measure the goodness of fit of estimated to true kernels we scaled them, in proportion to their mean contribution to the response, and computed the MSE between them (Section 2.3.8). The MSE between scaled true and estimated kernels was 5.6E-04 for the first-order kernel (Fig. 2.13), and 1.42E-03 for the second-order kernel slice with respect

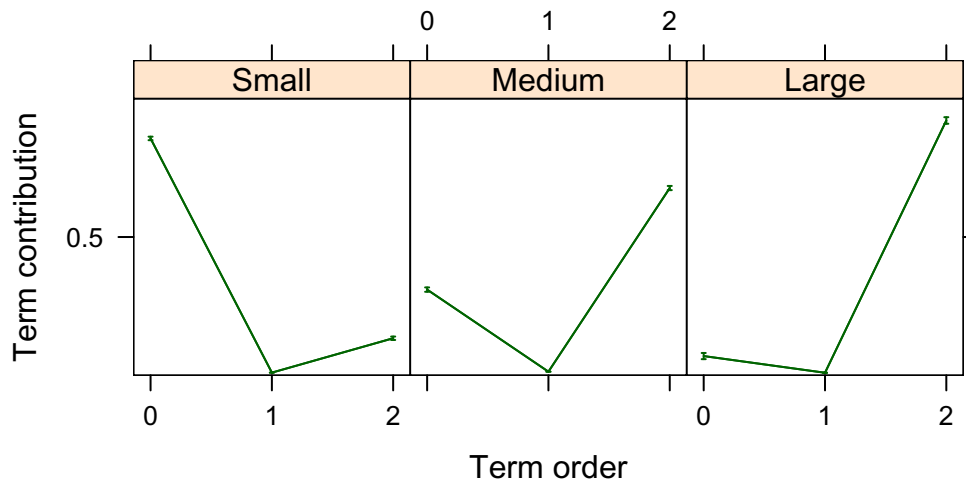


Figure 2.12: Simulated complex cell: mean relative contributions of Volterra terms to the response. Contributions have been averaged over different response ranges: (a) small responses (first quartile), (b) medium responses (second and third quartiles), and (c) large responses (fourth quartile). Medium and large responses of the purely-quadratic complex cell model are dominated by the second order Volterra term. The contributions of the zeroth-order term reflect a spurious fit (see text).

to Position (6, 6) (Fig. 2.14). This slice was the one with maximum MSE among all slices (the mean MSE across all slices was 3.06E-04). Thus, despite the noise, the estimated kernels accurately approximate their true values.

Note that for the complex cell, the first-order kernel should be zero. The estimated kernel is not zero and has structure. This is because in the linear Volterra equation, (2.10), small first-order coefficients, corresponding to a first-order Volterra term with small contribution (Fig. 2.12), will be dominated by large coefficients, corresponding to spontaneous and second-order Volterra terms with large contributions. Then, in the regression procedure, the small first-order coefficients will tend to fit noise. Subsequently, in the reconstruction of the estimated first-order kernel, (2.15), these noisy first-order coefficients will multiply relevant dimensions with clear structure, generating first-order kernels with spurious structure. Consequently, we should not try to interpret kernels corresponding to terms whose contributions to the response is very small.

2.4.2.2 PPR versus STC

The last section showed that PPR is a good technique to estimate relevant dimensions to use in the construction of Volterra models of simulated complex cells. Here, we compare PPR with the modification of STC for natural images proposed by Touryan et al. (2005), henceforth referred as STC for natural images. These authors mentioned that natural images are highly variable in their global contrast, *i.e.*, variance. To avoid possible contrast adaptation of the cells, they normalized the natural images to have equal variance. Our simulated complex cell did not include adaptation mechanisms, so we did not need to normalize the images to have equal variance. When we used STC for natural images with simulated responses to non-equal-variance images, we obtained poor results. But when we used responses to equal-variance images we obtained satisfactory results. To alleviate this problem, we equalized the variance of the natural images as a pre-processing step before computing STC. However, in these situations, we did not modify the simulated responses. The variance-equalization step was only necessary for the estimation of STC

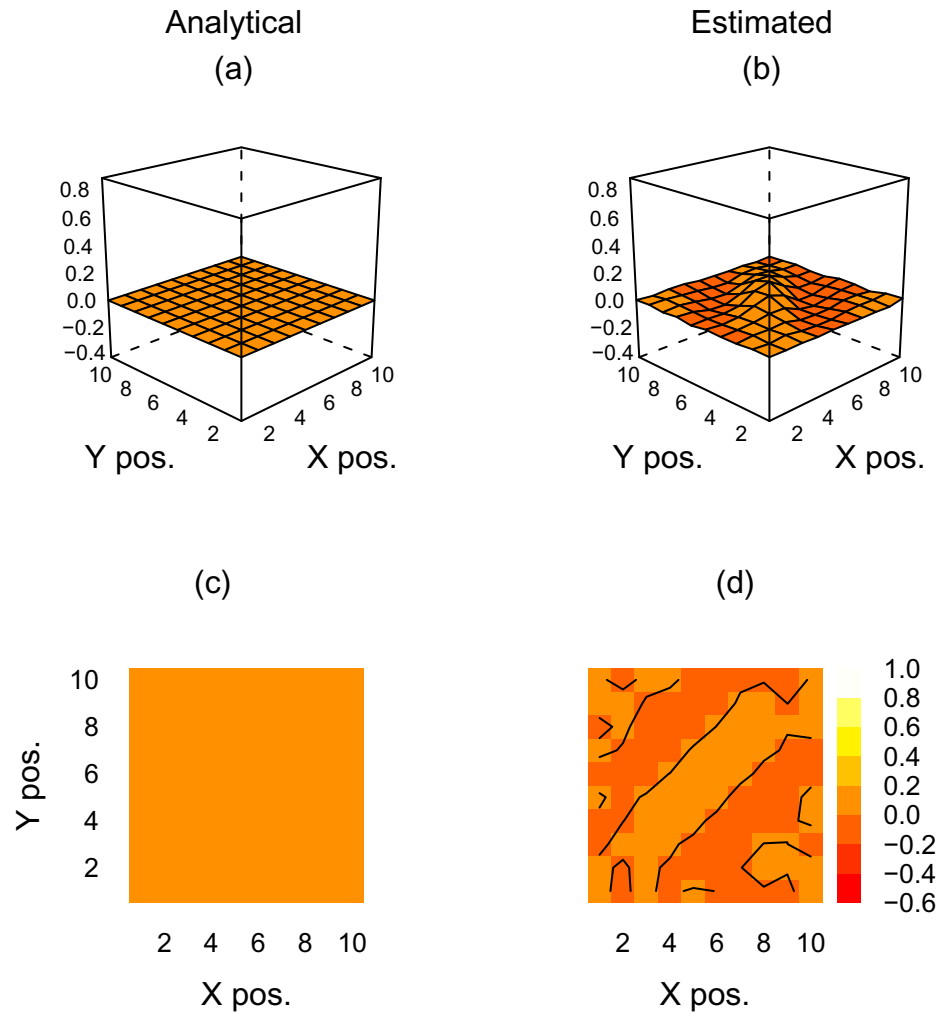


Figure 2.13: Simulated complex cell: first-order kernels. (a), (c): true. (b), (d): estimated. (a), (b): perspective plot. (c), (d): contour plot. The kernels have been scaled to reflect their mean contribution to the cell response. Although the estimated but not the true kernel has some structure, the amplitude is low, making the estimation not much different from zero, $MSE=5.6E-04$.

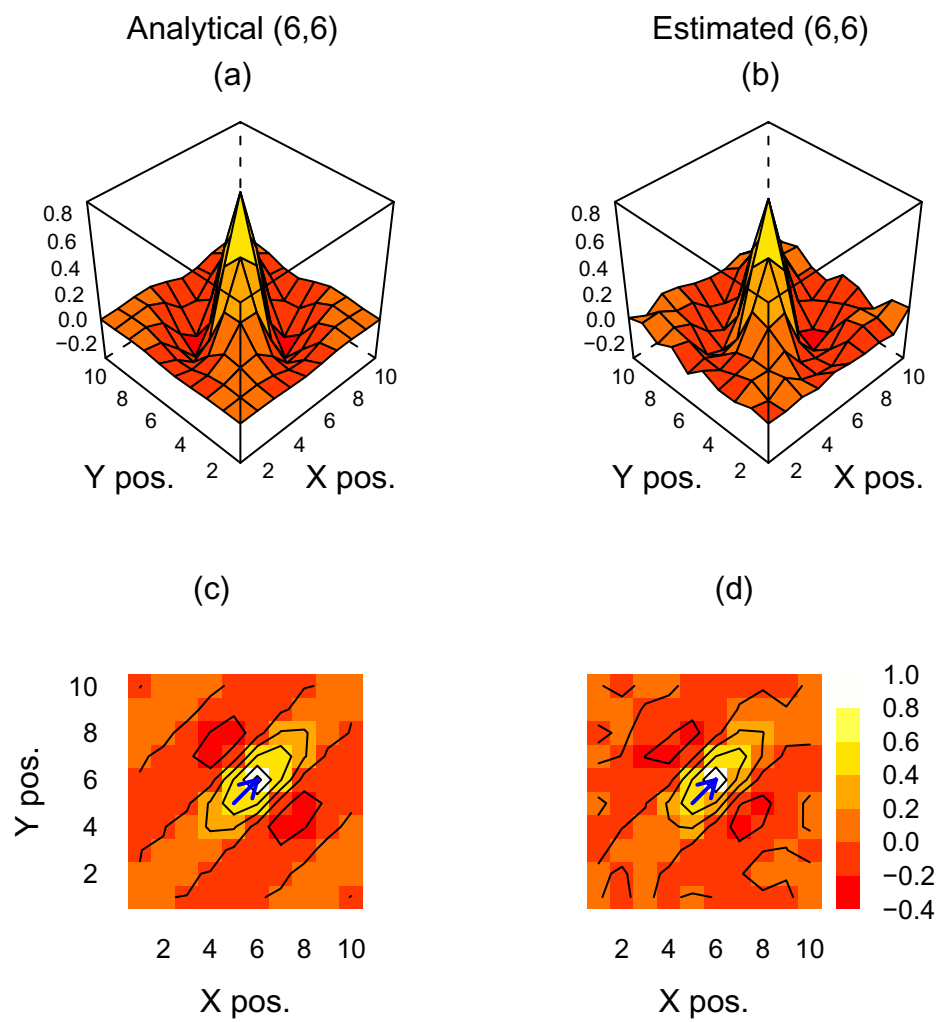


Figure 2.14: Simulated complex cell: second-order kernel slices with respect to Position (6, 6). (a), (c): true. (b), (d): estimated. (a), (b): perspective plot. (c), (d): contour plot. Blue arrows in the contour plots indicate the reference position. Despite the noise, the estimated kernel accurately approximates its true values, $MSE=1.42E-03$.

for natural images. Having estimated the relevant dimensions, the original, non-equal variance natural images were used to recover the nonlinearity using histograms.

Fig. 2.15a shows the average r^2 statistic as a function of the number of images for several conditions involving PPR and STC. Fig. 2.15b plots the average r^2 statistic as a function of the average number of spikes in the responses. STC achieves its best performance with responses to natural images with equal variance (red curve), as in Touryan et al. (2005). When the responses correspond to non-equal-variance images, STC for natural images performs poorly (blue curve) and the variance-equalization pre-processing step significantly improves its performance (green curve). With non-equal-variance data, PPR outperforms STC for natural images for any number of inputs or signal-to-noise ratios (black versus green and blue curves). However, STC has an advantage in running time over PPR. The latter is a time consuming iterative algorithm, requiring several estimations of smooth functions and least-squares minimizations. STC is a fast algorithm that only requires the computation of covariance matrices and their spectral decomposition. A fast algorithm was essential for the online estimation of the relevant dimensions of a cell, an impressive task performed by Touryan et al. (2005).

2.4.2.3 Volterra model versus histogram-based model

Section 2.4.2.1 showed that Volterra models can represent the nonlinearities of complex cells well, starting from its estimated relevant dimensions. Here, we compared the performance of the Volterra model with that of a complex-cell histogram-based model (Section 2.3.4.5). To make the results independent from the method used to estimate the relevant dimensions, we compare the Volterra model using the relevant dimensions estimated by PPR, the Volterra model using the STC relevant dimensions, the histogram-based model using the PPR relevant dimensions, and the histogram-based model using the STC relevant dimensions. Fig. 2.16a and Fig. 2.16b plot the correlation coefficient between the simulated cell responses and the models predictions as a function of the number of inputs and the number of spikes in the simulated responses, respectively. The relevant

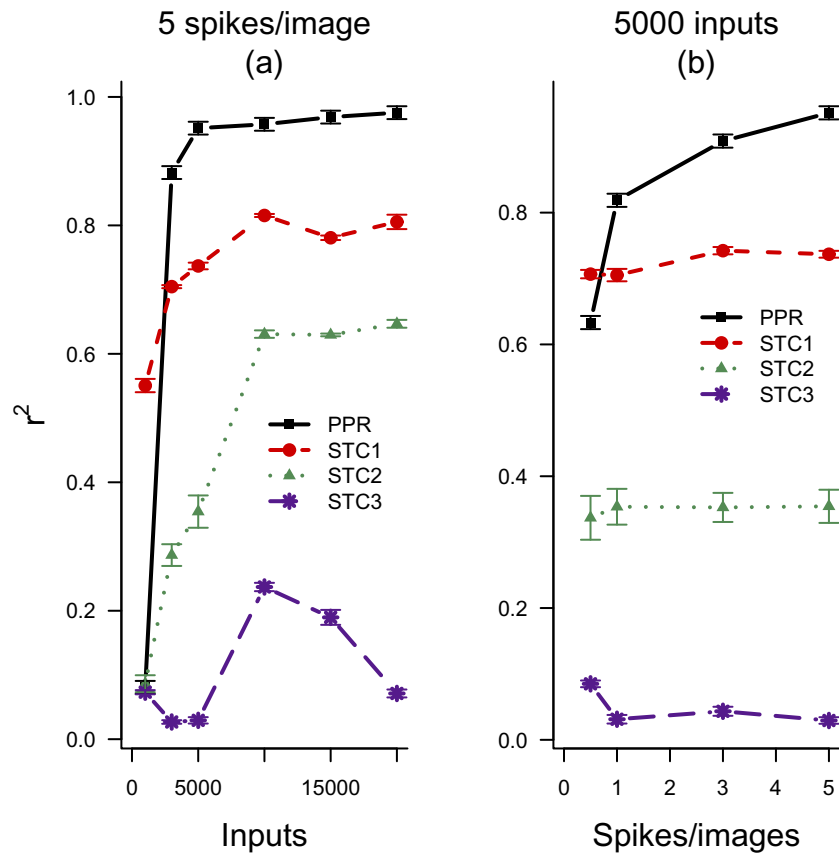


Figure 2.15: Simulated complex cell: comparison between PPR and STC for natural images at various conditions. These conditions were the following: Red curve: STC1, equal-variance responses; Green curve: STC2, non-equal-variance responses with variance-equalization pre processing; Blue curve: STC3, non-equal-variance responses without variance-equalization pre processing. The black curve is for PPR with non-equal-variance responses. The figure shows average r^2 statistic for the two relevant dimensions of the simulated complex cell. This statistic appears as a function of (a) the number of inputs, when the mean response is 5 spikes/image, and (b) the number of spikes per image, when using 5,000 image-response pairs. The size of the error bars is two standard errors. STC achieves its best performance for responses to equal-variance images. Variance-equalization pre processing improves the performance of STC when using responses to non-equal-variance images. For responses to non-equal variance images, PPR outperforms STC for any number of inputs and signal-to-noise ratios.

dimensions estimated by PPR yield better predictions than those estimated by STC, for both the Volterra (black versus red curves) and the histogram-based models (blue versus green curves). When using the PPR relevant dimensions, the Volterra model predicts responses slightly better than the histogram based model (black versus blue curves). With the STC relevant dimensions, both models perform similarly (red versus green curves).

2.4.3 Cortical Complex Cell

Our results with simulated data from a cortical complex cell have shown that PPR can effectively estimate the cell's true relevant dimensions. Moreover, these results showed that the Volterra relevant-space technique can accurately estimate the cell's true Volterra kernels. Finally, the results indicated that a Volterra model accurately predicts the simulated cell's responses. However, these results cannot tell whether our method is practical for real cells. Here, we test PPR, STC, the Volterra model, and the histogram-based model with physiological data from a cortical complex cell obtained by Touryan et al. (2005).

As for the simulated data, we began by estimating the mean predictive error of PPR models as a function of the number of estimated relevant dimensions as shown in Fig. 2.2c. Based on this figure, we used three relevant dimensions for the Volterra model. The relevant dimensions estimated by PPR were noisy. To smooth them, we applied a low pass filter. We then orthonormalized them. The resulting relevant dimensions used to estimate the Volterra model are shown in Fig. 2.17. They have clear structure, showing a 45° orientation preference. These relevant dimensions yielded better predictive values than the original noisy relevant dimensions.

As before, we selected the order of the Volterra model as the least order maximizing the average predictive power. Fig. 2.4c shows the mean predictive power of Volterra models as a function of their order. Based on this data, we selected a second-order Volterra model for the cortical complex cell. As for the simulated complex cell, we determined the relative contributions of the various Volterra terms to the response of the cortical complex cell.

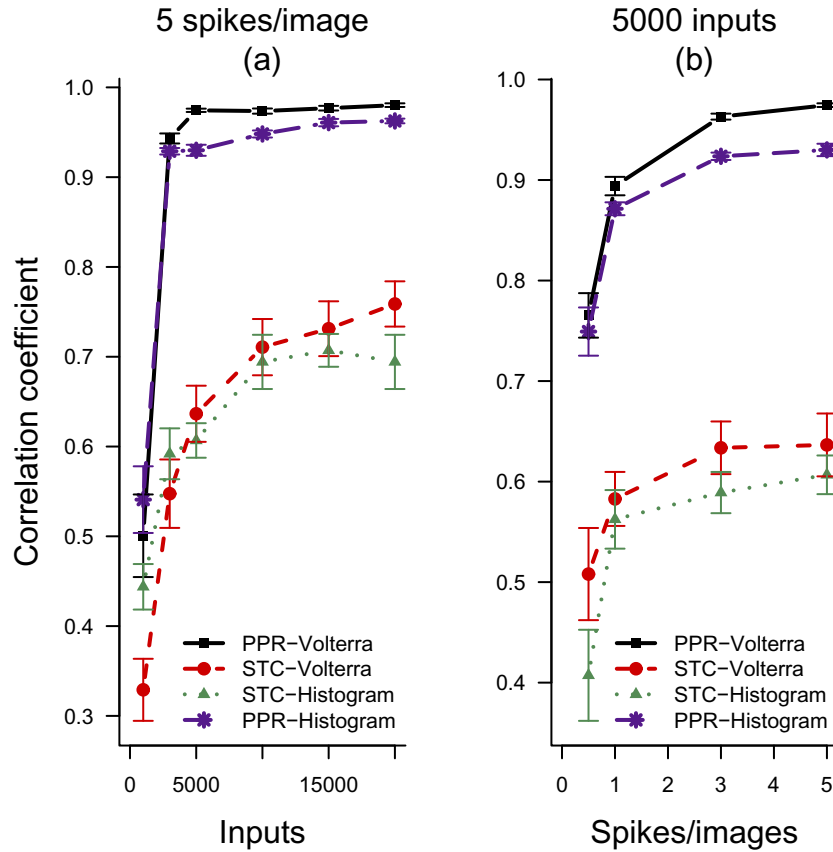


Figure 2.16: Simulated complex cell: predictions of the Volterra model and the non-linearity estimated from histograms. This figure shows average correlation coefficients between simulated cell responses and model predictions. These coefficients appear as functions of (a) the number of inputs when the mean responses was 5 spikes/image, and (b) the number of spikes per image, when using 5,000 image/response pairs. Black curve: Volterra model using the PPR relevant dimensions. Red curve: Volterra model using the STC relevant dimensions. Green curve: nonlinearity estimated from histograms using the STC relevant dimensions. Blue curve: Nonlinearity estimated from histograms using the PPR relevant dimensions. The size of the error bars is two standard errors. The relevant dimensions estimated by PPR yield better predictions than those estimated by STC. When using the PPR relevant dimensions, the Volterra model predicts responses slightly better than the histogram-based model.

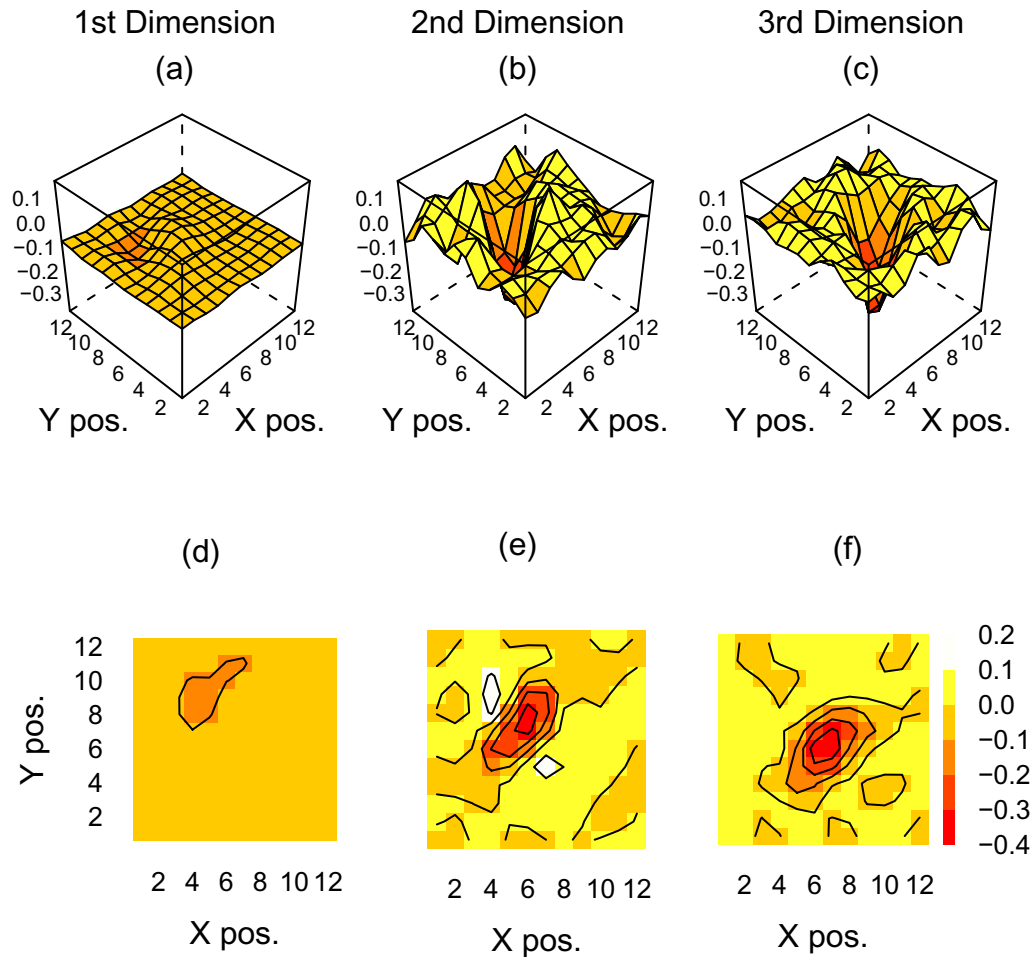


Figure 2.17: Cortical complex cell: PPR relevant dimensions. (a), (d): first relevant dimension. (b), (e): second relevant dimension. (c), (f): third relevant dimension. (a), (b), (c): perspective plot. (d), (e), (f): contour plot. The relevant dimensions have clear structure, showing a 45° orientation preference.

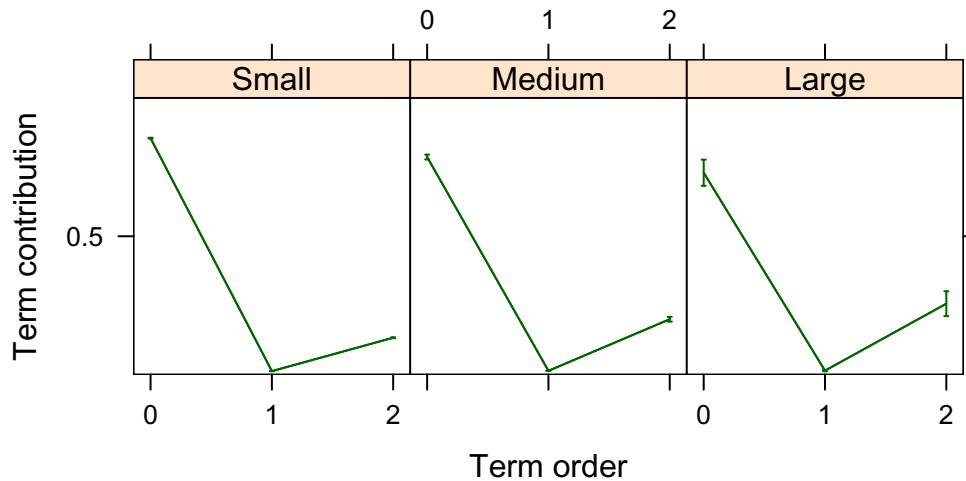


Figure 2.18: Cortical complex cell: mean relative contributions of Volterra terms to the response. Contributions have been averaged over different response ranges: (a) small responses (first quartile), (b) medium responses (second and third quartiles), and (c) large responses (fourth quartile). This complex cell has a large spontaneous activity. Relative contributions from the first-order term are negligible. As the response level increases relative contributions from the spontaneous term become smaller, while relative contributions from the second-order term become larger.

We averaged the contributions across small responses (first quartile), medium responses (second and third quartiles), and large responses (fourth quartile). These contributions are shown in Fig. 2.18. At all response levels, the term with most relative contribution was the spontaneous one, indicating that the complex cell had a strong background activity. Also, relative contributions from the first-order term are negligible, as for the simulated complex cell (Figure 2.12). As the response level increased, relative contributions from the spontaneous term became smaller, while contributions from the second-order term increased.

The left column of Fig. 2.19 shows the estimated first-order kernel and the right column the estimated second-order kernel slice with respect Position (5, 9). This was the second order kernel slice with largest value. Both kernels have been scaled to reflect their mean contribution to the cell response. Because contributions from the first order Volterra

term are negligible (Fig. 2.18), as in the case of the simulated complex cell, the first order kernel is probably spurious (see discussion at the end of Section 2.4.2.1). Thus, we refrain from interpreting this kernel. The second-order kernel slice with respect to Position (5, 9) is positive everywhere, being larger in the vicinity of the reference position (blue arrow) and smaller at positions along the central diagonal. Thus, positive correlations between a point of light at the reference position and a point of light in its close surrounding will induce the strongest pairwise response facilitation. In turn, positive correlations between a point of light at the reference position and a point of light in the central diagonal (along the anti-preferred orientation) will induce weaker facilitation.

Points outside and near the boundary of the receptive field should not modulate the response in any sense, either linearly or nonlinearly. For the linear kernel, we observe that points near the boundary of the receptive field are close to zero. This is what we would expect from a well estimated linear kernel. However, the second-order kernel slice is positive everywhere. This illustrates a problem, because it shows that points on the boundary of the receptive field nonlinearly modulate the cell response. This problem could be an artifact of over smoothing the PPR relevant dimensions. However, these possibly over-smoothed relevant dimensions lead to better predictions than relevant dimensions with less smoothing (including non-smoothed relevant dimensions). An alternative explanation is that the nonlinear component of the receptive field could be larger than its linear component. The size of the images used in the physiological experiments could have been adequate to characterize the linear component of the receptive field, as illustrated in Figures 2.19a and 2.19c, but not large enough to characterize the nonlinear component, Figures 2.19b and 2.19d.

Because the cortical complex cell's responses are dominated by the second-order kernel, it is now presented in greater detail. Fig. 2.20 shows estimated second-order kernel slices with respect to Positions (7, 7) and (9, 5). The former slice (Fig. 2.20, left column) is nearly flat and contains the smallest pairwise facilitation among all second-order

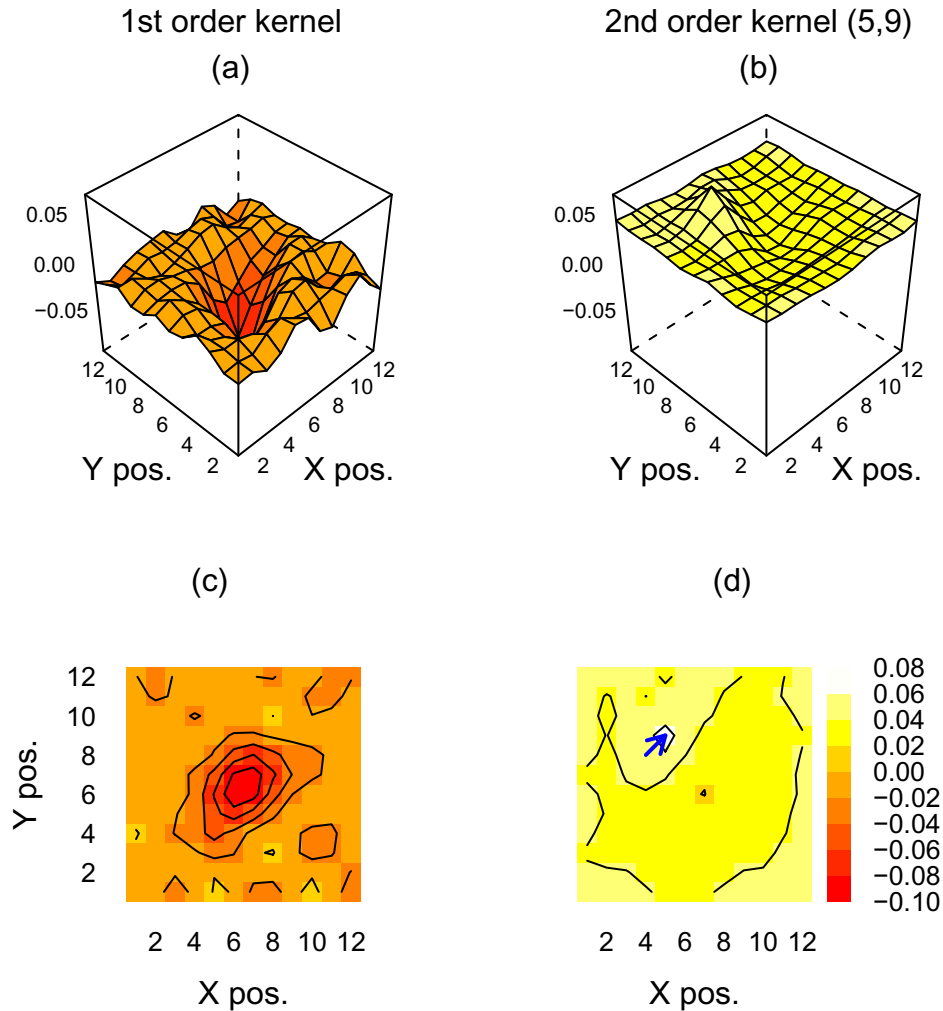


Figure 2.19: Cortical complex cell: first and second-order kernels. (a), (c): first order. (b), (d): second-order kernel slice with respect to Position (5, 9). (a), (b): perspective plot. (c), (d): contour plot. The blue arrow in (d) points to the reference position of the second-order kernel slice. Because contributions from the first order Volterra term are negligible (Fig. 2.18), as in the case of the simulated complex cell, the first order kernel is probably spurious. The second-order kernels slice indicates that positive correlations between a point of light at the reference position and a point of light in its close surrounding will induce the strongest pairwise response facilitation. In turn, positive correlations between a point of light at the reference position and a point of light in the central diagonal (along the anti-preferred orientation) will induce weaker facilitation.

slices. For the latter slice (Fig. 2.20, right column) facilitation is strongest for interactions between the reference position and points in its close vicinity, or points in the brightest region on the top-left quadrant. Facilitation is weaker for interactions between the reference position and points along the central red band.

We next performed an STC analysis on the responses of the cortical complex cell and selected its first two relevant dimensions to describe the relevant space. As described in Section 2.3.4.4, we regularized the STC relevant dimensions by only normalizing the eigenvectors corresponding to the 35% largest eigenvalues. These relevant dimensions are shown in Fig. 2.21. They show a cleaner structure than that of the PPR relevant dimensions in Fig. 2.17. However, best relevant dimensions are those that lead to better predictions, which we discuss next.

To compare the performance of the Volterra model with that of the complex-cell histogram model, and to make the results independent of the algorithm used to estimate the relevant dimensions, we evaluated the predictive power of both models with PPR and STC relevant dimensions. Fig. 2.22 shows average cross-correlation coefficients between the predictions of the models and the complex-cell responses as a function of the number of inputs. Note that, as discussed in Sec. 2.3.3, the data used to evaluate the different models was different from that used to estimate their parameters. Thus, the predictive power measures are not inflated by over-fitting effects.

For any number of inputs, the Volterra models yields better predictions than the histogram based model with both the PPR relevant dimensions (black versus blue curves) and the STC relevant dimensions (red versus green curves). For more than 1,000 inputs, the PPR relevant dimensions produced equal or better predictions than the relevant dimensions estimated by STC, for both the Volterra (black versus red curve) or histogram-based (blue versus green curve) models. These results are not due to a faulty implementation of the STC algorithm or histogram based models. The correlation coefficient of the histogram-based model estimated from 20,000 images using the STC relevant dimensions lies in the upper range of those reported in Figure 6 of Touryan et al. (2005).

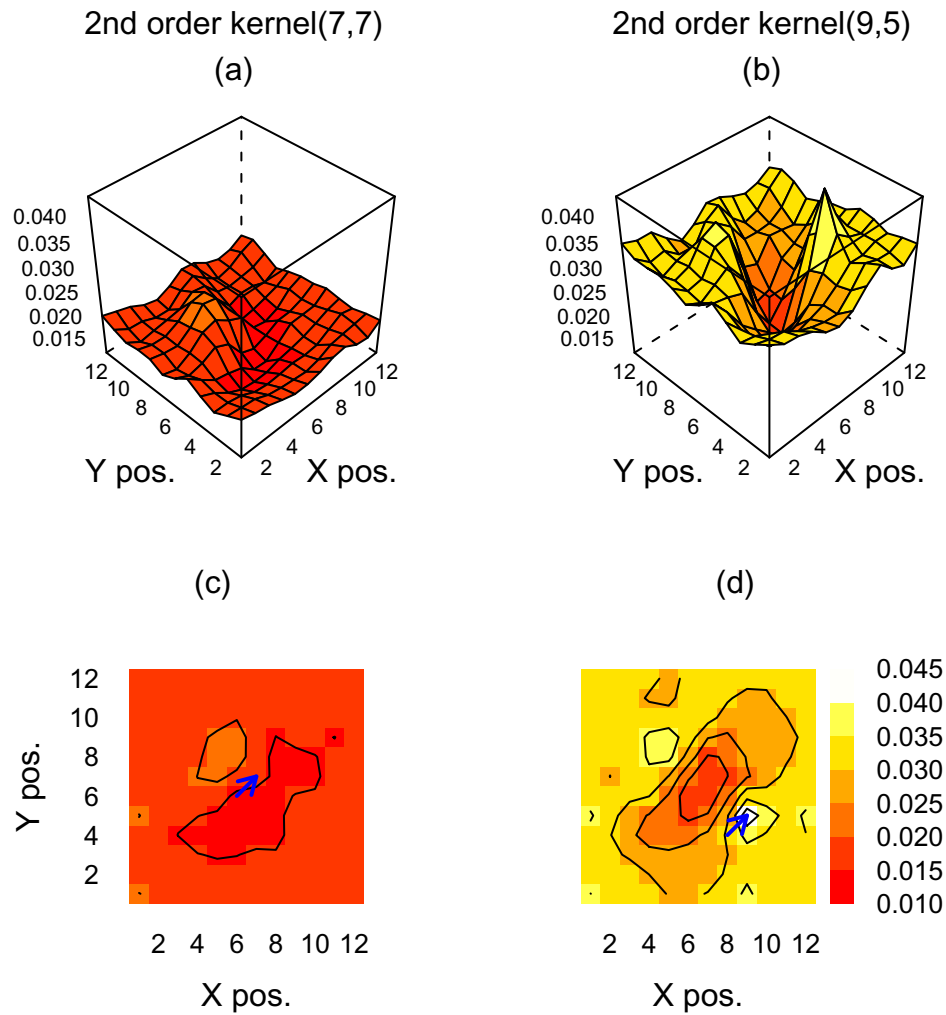


Figure 2.20: Cortical complex cell: second-order kernel slices estimated from PPR relevant dimensions. Slices correspond to different reference positions. (a), (c): Position (7, 7); (b), (d): Position (9, 5). Blue arrows in the contour plots indicate the reference positions. The slice with respect to Position (7, 7), left column, is nearly flat and contains the smallest pairwise facilitation among all second-order slices. For the slice with respect to Position (9, 5), right column, facilitation is strongest for interactions between the reference position and points in its close vicinity, or points in the brightest region on the top-left quadrant. Facilitation is weaker for interactions between the reference position and points along the central red band.

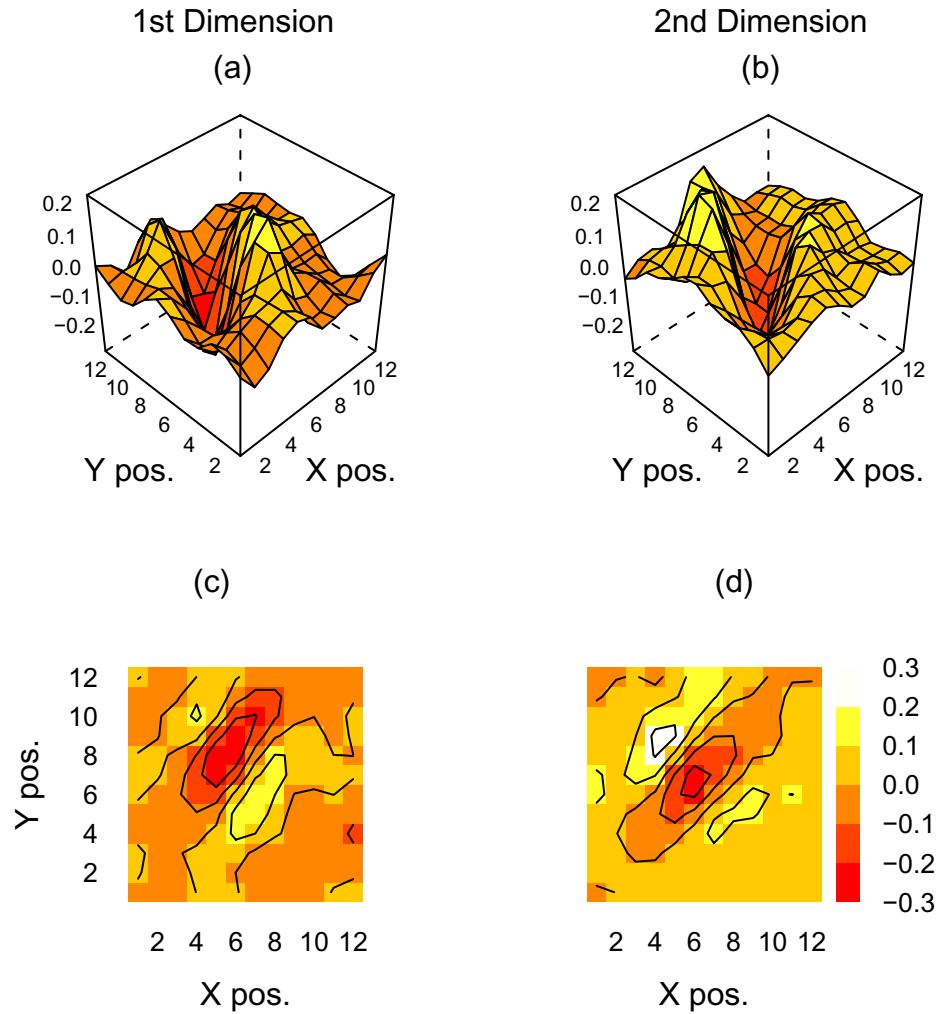


Figure 2.21: Cortical complex cell: STC relevant dimensions. (a), (c): first relevant dimension. (b), (d): second relevant dimension. (a), (b): perspective plots. (c), (d): contour plots. These relevant dimensions show a cleaner structure than that of the PPR relevant dimensions in Fig. 2.17. However, best relevant dimensions are those that lead to better predictions (see text).

Hence, of the models tested, we found that the best combination is to use Volterra models with PPR relevant dimensions.

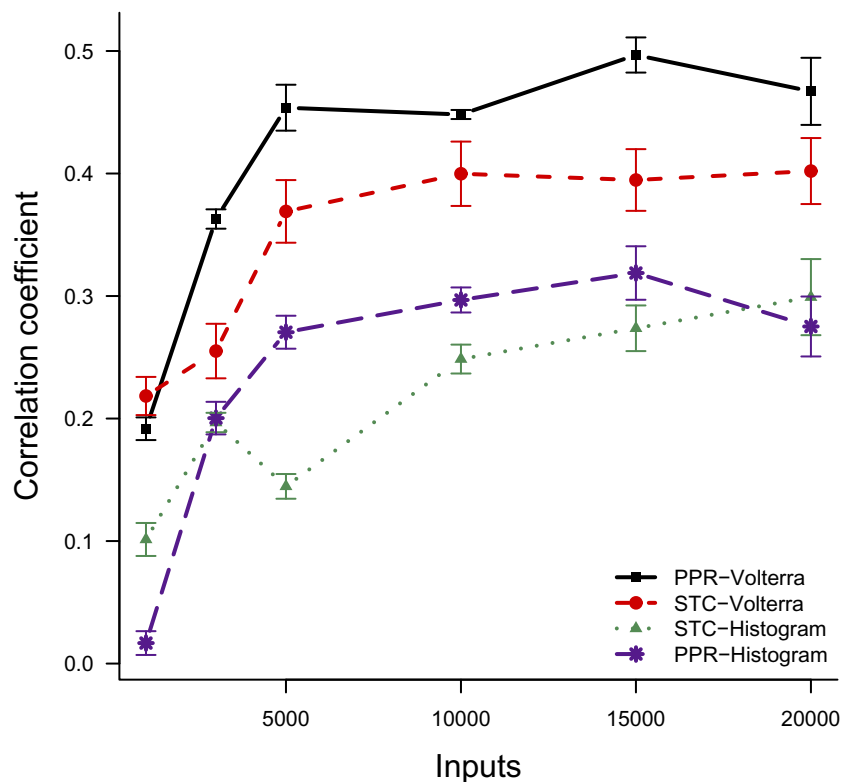


Figure 2.22: Cortical complex cell: predictions with two classes of models and two classes of estimated relevant dimensions. This figure shows average correlation coefficients between cell responses and model predictions. These coefficients appear as functions of the number of inputs. Black curve: Volterra model using the PPR relevant dimensions. Red curve: Volterra model using the STC relevant dimensions. Green curve: Nonlinearity estimated from histograms using the STC relevant dimensions. Blue curve: Nonlinearity estimated from histograms using the PPR relevant dimension. The size of the error bars is two standard errors. For any number of inputs, the Volterra models yields better predictions than the histogram based model with both the PPR relevant dimensions (black versus blue curves) and the STC relevant dimensions (red versus green curves). For more than 1,000 inputs, the PPR relevant dimensions produced equal or better predictions than the relevant dimensions estimated by STC, for both the Volterra (black versus red curve) or histogram-based (blue versus green curve) models.

2.5 Discussion

Receptive fields are the main theoretical framework to represent functional properties of visual cells. In the most standard form, receptive fields tell how a cell responds when a point of light falls in a position of space (or time). Therefore, traditionally, one can think of receptive fields as spatial (or temporal) impulse responses, *i.e.*, we tend to think of receptive fields as linear filters. In addition, the majority of stimuli used to obtain receptive fields are artificial, *e.g.*, dots of light, oriented bars, or sinusoidal gratings. Receptive fields obtained with such stimuli have been useful. However visual cells are often nonlinear and their functional properties may depend on the stimulus used to map them. In that case, we would like to map nonlinear receptive fields with natural images. How could receptive-field properties depend on the stimulus used to map them? One reason is that sensory systems adapt to the environment, changing their own functional properties (see Grzywacz & Balboa, 2002, and references therein). Another reason is that, as demonstrated by Proposition 2, by using a given stimuli set to characterize a cell, we can only observe the projection of the cell kernels onto the relevant space defined by the statistics of the stimulus set. In other words, the stimuli used to probe a cell constrains the functional properties that can be observed from its responses. We thus searched for a model that would allow us to map nonlinear receptive fields with natural stimuli. We explored the Volterra model, as it encompasses the receptive field as the first-order kernel and generalizes receptive fields to the nonlinear domain.

The large number of parameters in spatial Volterra models has hindered their application in the field of visual Neuroscience. Fortunately, responses of visual cells appear to depend on a very low-dimensional subspace of the image space (Section 2.2.2). Consequently, we could introduce the Volterra relevant-space technique for Volterra models. This representation allows us the estimation of Volterra models from natural images.

To test the Volterra relevant-space technique, we used data from simulated cortical simple and complex-cell models, for which we could analytically derive the true Volterra

kernels. The complex cell model, by being purely quadratic, can be exactly represented by a finite-order Volterra model. This is not the case for the simple-cell model, for which Volterra kernels of any order are non-zero. (However, when the projections of the input images onto the relevant dimensions are not too large, the Volterra model can be a good approximation for the simple-cell model. This would be true for natural images, which tend to have low contrasts – (Balboa & Grzywacz, 2000; Tadmor & Tolhurst, 2000; Ruderman & Bialek, 1994; Zhu & Mumford, 1997)) Thus, the simple-cell model is a challenging test for the Volterra relevant-space technique. For both models, we showed that estimated kernels accurately approximate their true counterparts (Figs. 2.6, 2.7, 2.13, and 2.14), illustrating that the Volterra relevant-space technique can accurately estimate Volterra models.

We also applied the Volterra relevant-space technique to physiological data from cortical complex cells. The estimated Volterra models yielded predictions of responses that were superior to those of a histogram-based model (Figure 2.22). This establishes this technique as a viable alternative for the estimation of nonlinear receptive fields with natural images.

An important step in the Volterra relevant-space technique is the estimation of relevant dimensions. We explored here three methods for this determination, namely, PPR, STA, and STC. For simulated data from cortical simple- (Fig. 2.8) and complex-cells (Fig. 2.15) we found that the relevant dimensions estimated by PPR better approximated the true relevant dimensions of the cell models. For physiological data from cortical complex cells, the relevant dimensions estimated by PPR yielded equal or better predictions than those estimated by STC (Fig. 2.22). Furthermore, based on the simulated complex-cell data, PPR will probably succeed in estimating relevant dimensions for non-equal variance responses, while STC will likely fail. In contrast, a disadvantage of PPR is its running time. PPR is a time-consuming iterative algorithm, requiring several estimations of smooth functions and least-squares minimizations. STA and STC are comparatively fast algorithms that only require matrix operations.

We also showed that the Volterra model has advantages over histogram-based Models for representing nonlinearities. For simulated data from simple and complex cells, we observed that, when using PPR relevant dimensions, the Volterra model could predict responses significantly better than the histogram-based models. When using the STA or STC relevant dimensions the Volterra model performs equally as the histogram based models (Fig. 2.9, Fig. 2.16). For physiological data from cortical complex cells, the Volterra model predicted responses better than a histogram-based model when using either PPR or STC relevant dimensions (Fig. 2.22). In summary, the Volterra model predicts responses equally or better than histogram-based models. A further advantage is that after the relevant dimensions have been estimated, the Volterra relevant-space technique uses only linear operations. Thus, it is fast and easy to implement. In contrast, most implementations of histogram-based techniques require nonlinear optimizations that are slower and more subtle to implement.

From a prediction point of view, Volterra models are very accurate. However, a note of caution is pertinent regarding the interpretation of estimated Volterra kernels. Because Volterra models have a large number of parameters, physiological data is limited in size, and neurons are noisy, least-squares fits to Volterra models can achieve similar predictive values with qualitatively different parameters. This could happen even after using different data sets to estimate a model and evaluate its predictive power. To minimize the effect of this problem, we have adopted several caution measures. First, we always use separate evaluation and test data sets (Section 2.3.3). Second, we divide our evaluation data set into jackknifed data sets (Efron & Tibshirani, 1993), estimate separate parameters (relevant dimensions and Volterra coefficients) for each jackknifed data set, and average our estimates. Third, we evaluate the relative contributions of Volterra terms (Figures 2.5, 2.12, 2.18), and we do not try to interpret kernels of terms with low relative contributions (see justification in last paragraph of Section 2.4.2.1). Fourth, whenever possible, we compare Volterra kernels estimated from different sets of relevant dimensions.

In this paper, we have contrasted Volterra and histogram-based models. However, these two types of models are complementary, each describing the nonlinearity in a different way. For one- or two-dimensional relevant spaces, histogram-based models provide a succinct way of visualizing the nonlinearity. The nonlinearity would depend on the projection of the input images on one or two relevant dimensions. On the other hand, Volterra models can be estimated from many more than two dimensions and show more explicitly how individual pixels in the image contribute, linearly and nonlinearly, to generate responses. This is more in line with conventional models of receptive fields and thus, may facilitate mechanistic interpretations for the cells. This mechanistic view of the nonlinearity could be of interest to many physiologists. For example, in the retina, starburst amacrine cells may produce a lateral nonlinear facilitation of the responses of certain ganglion cells. This facilitation would be mediated by acetylcholine (ACh) (Tauchi & Masland, 1984; Amthor, Grzywacz, & Merwine, 1996). To validate this hypothesis, one could estimate Volterra models of the appropriate ganglion cells, while blocking ACh and in control condition. The prediction would be that a nonlinear Volterra kernel would lose a central facilitatory zone while blocking ACh.

2.5.1 Relation to Previous Research

Wiener series predated Volterra models for the characterization of nonlinear physiological systems. Wiener developed this series as an orthogonal representation of Volterra series for white noise inputs (Wiener, 1958). Later Lee and Schetzen (1965) introduced an efficient method to estimate Wiener series, the cross-correlation technique. This technique allowed many investigators to use Wiener series to study visual neurons (P. Marmarelis & Naka, 1972; Victor & Shapley, 1980; Emerson, Citron, Vaughn, & Klein, 1987; Livingstone & Conway, 2003). Wiener series have several analytical and practical advantages over Volterra series, and the cross-correlation technique greatly simplifies their estimation. However, Wiener series have the following three major limitations for the characterization of visual cells from natural images, namely: First the orthogonality properties of Wiener

series and the cross-correlation technique apply only to Gaussian white noise inputs. Second, Wiener kernels lack the relatively simple interpretations of Volterra kernels. Third, the kernels estimated by cross-correlation have large estimation variance (V. Z. Marmarelis, 2004).

These limitations motivated the development of other methods for the estimation of Volterra models. Their main problem is the exuberance of parameters, so dimensionality-reduction techniques are needed to make the estimation of Volterra models feasible. The *kernels-expansion technique*, initially proposed by Wiener for the estimation of Wiener kernels, has been the method of choice for the estimation of Volterra models (Bose, 1956; Lee, 1964; Amorocho & Brandstetter, 1971; Watanabe & Stark, 1975; V. Marmarelis, 1993). It reduces the number of parameters by compacting the representation of kernels and avoids the computation of cross-correlations, performing instead least-squares fitting of the actual data to model parameters. The representation of Volterra kernels is compacted by expressing them as linear combinations of a few basis functions. One then uses the compact representation to build a low-dimensional Volterra model. Then, the kernels of the original Volterra model are reconstructed from the estimated parameters of the low-dimensional Volterra model.

The critical step in the dimensionality reduction of Volterra models for visual cells is the selection of basis functions. This is critical, because they could bias the estimated kernels. For example, first-order Volterra kernels can only be linear combinations of the selected basis functions, as shown in (2.15). Therefore, an improper selection of basis functions will impede a correct estimation of Volterra kernels. This error will be independent of the amount and quality of data used in the estimation procedure.

Marmarelis used Laguerre basis functions to analyze systems excited by 1D, temporally modulated, inputs (V. Marmarelis, 1993). One rationale for selecting such basis functions is that discrete Laguerre basis functions are complete and orthonormal. Furthermore, they have an exponential time decay. Such decay approximates well the known

linear kernels of many physiological systems with 1D, temporally modulated inputs. However, it is not clear whether higher-order Volterra kernels can be represented by similar basis functions as those for first-order Volterra kernels. Another issue is that, for every cell under study, the experimenter must select the number of Laguerre basis functions, *i.e.*, L in (2.15) and (2.16), to represent the kernels. The experimenter must also select the parameter α of the Laguerre basis functions. Marmarelis (1993) recommended a judicious selection of L and α based on trial and error. Later, Mistis and Marmarelis (2002) proposed a method to learn the parameter α from input/output data.

For systems excited by 2D inputs, *e.g.*, visual cells, 1D Laguerre basis functions cannot be used to represent their 2D kernels. No known set of orthonormal 2D basis functions fulfills the role that Laguerre functions have for systems with 1D, temporally modulated, inputs. Moreover, the number of parameters needed to characterize 2D basis functions is larger than that required for 1D basis functions. Tuning these parameters thus becomes an arduous process in the 2D case. For systems excited by inputs varying in time and space, *i.e.*, 3D inputs, this problem is aggravated even further.

These difficulties in the selection of basis functions motivated our search for an alternate method to reduce the dimensionality of Volterra models. Instead of reducing the dimensionality of the unknown kernels of the Volterra model, we thought that a better approach might be to reduce the dimensionality of the input images. This latter reduction leads to the Volterra relevant-space technique. The Volterra relevant-space technique has the following three advantages with respect to the kernels-expansion technique:

1. The effectiveness of the low-dimensional representation of Volterra models depends upon using a small set of basis functions. In the kernels-expansion technique, the assumption for effectiveness is that the kernels of the Volterra model can be represented using a small set of basis functions. This assumption cannot be tested experimentally *a priori*. In the Volterra relevant-space technique, the assumption is that, for a given cell, the relevant-space is low-dimensional. As demonstrated in

previous studies (de Ruyter van Steveninck & Bialek, 1988; Sharpee et al., 2006; Rust et al., 2005), this assumption has experimental support.

2. Optimization of (2.8) allows the Volterra relevant-space technique to estimate the relevant dimensions from the data. Consequently, the Volterra relevant-space technique avoids the bias introduced in the estimates of kernels by an improper, ad hoc selection of basis functions.
3. As described in Section 2.3.4.1, the solution of (2.8) is a well-studied problem in different fields of science. We can then use the expertise from these field in the estimation of Volterra models.

2.5.2 Limitations of Volterra Models

Although one can use Volterra models to study receptive fields, they have limitations in their applicability. We list the following three:

1. Convergence of Volterra series requires certain smoothness conditions on the function being approximated (Palm & Poggio, 1977). This requirement is not severe, since the firing rate of visual cells can be adequately approximated by smooth functions.
2. The response could depend on very high-order Volterra terms. Consequently, even using the Volterra relevant-space technique, we might not be able to estimate them from a limited amount of physiological data. However, because many cells in the early visual system have been adequately characterized with linear or quadratic models, this scenario does not seem a strong limitation. Furthermore, studies of statistics of natural images find that distributions of variables that modulate responses tend to have maxima at values that generate little response. For example, the distribution of contrasts tends to be maximal at zero (Balboa & Grzywacz, 2000; Tadmor & Tolhurst, 2000; Ruderman & Bialek, 1994; Zhu & Mumford,

1997). Therefore, responses to natural images may be low, generating relatively little high-order nonlinearities.

3. A cell response may admit a Volterra representation, with all relevant terms of moderate order, but it may not depend on a low-dimensional subspace of the image space. In such a case, the Volterra relevant-space technique will not provide substantial dimensionality reduction to allow the estimation of high-order Volterra models. However, physiological experiments with some cells in the early visual system come to our aid (de Ruyter van Steveninck & Bialek, 1988; Sharpee et al., 2006; Rust et al., 2005), showing that the responses of these cells depend on such a low-dimensional subspace.

2.5.3 Extension to the Temporal Domain

Although we focus this paper on spatial Volterra models, extending the Volterra relevant-space technique to include time is theoretically not a limitation. We just need to add the temporal dimension to the Volterra model (2.3), yielding

$$\begin{aligned}
 y_{Q,k}(\mathbf{x}) &= k_0(t) + \sum_{t=1}^{N_t} \sum_{i,j=1}^{N_s} k_1(i,j,t)x(i,j,t) + \dots + \\
 &+ \sum_{t_1,\dots,t_Q=1}^{N_t} \sum_{i_1,j_1,\dots,i_Q,j_Q=1}^{N_s} k_Q(i_1,j_1,t_1,\dots,i_Q,j_Q,t_Q)x(i_1,j_1,t_1)\dots x(i_Q,j_Q,t_Q) + \varepsilon
 \end{aligned} \tag{2.26}$$

The Volterra relevant-space technique can also be extended to include the temporal dimension. However, the estimation of relevant dimensions in spatio-temporal models becomes a higher-dimensional problem and consequently, a more difficult one.

2.6 Appendices

2.6.1 Proofs

Proposition 1. Given $y_{Q,k}(\mathbf{x}) : \mathbb{R}^{N \times N} \rightarrow \mathbb{R}$, as in (2.3). If a vector space \mathcal{S} , spanned by a set of basis functions $\{\mathbf{b}_1, \dots, \mathbf{b}_L : \mathbf{b}_l \in \mathbb{R}^{N \times N}\}$, fulfills (2.8), then (2.3) can be expressed as (2.7), where q_0, \dots, q_Q are

$$q_0 = k_0 \tag{2.27}$$

$$q_1(l) = \sum_{r,s=1}^N k_1(r,s) b_l(r,s) \tag{2.28}$$

...

$$q_Q(l_1, \dots, l_Q) = \sum_{r_1, s_1, \dots, r_Q, s_Q=1}^N k_Q(r_1, s_1, \dots, r_Q, s_Q) b_{l_1}(r_1, s_1) \dots b_{l_Q}(r_Q, s_Q) \tag{2.29}$$

Proof. From (2.3), (2.5), and (2.8), observe that

$$\begin{aligned} y_{Q,k}(\mathbf{x}) &= y_{Q,k} \left(\sum_{l=1}^L \alpha_l(\mathbf{x}) \mathbf{b}_l \right) = k_0 + \sum_{r,s=1}^N k_1(r,s) \left(\sum_{l=1}^L \alpha_l(\mathbf{x}) b_l(r,s) \right) + \dots \\ &+ \sum_{r_1, s_1, \dots, r_Q, s_Q=1}^N k_2(r_1, s_1, \dots, r_Q, s_Q) \left(\sum_{l_1, \dots, l_Q=1}^L \alpha_{l_1}(\mathbf{x}) b_{l_1}(r_1, s_1) \dots \alpha_{l_Q}(\mathbf{x}) b_{l_Q}(r_Q, s_Q) \right) \\ &+ \varepsilon \end{aligned} \tag{2.30}$$

Switching the order of summations in (2.30), we see that

$$\begin{aligned} y_{Q,k}(\mathbf{x}) &= k_0 + \sum_{l=1}^L \left(\sum_{r,s=1}^N k_1(r,s) b_l(r,s) \right) \alpha_l(\mathbf{x}) + \dots \\ &+ \sum_{l_1, \dots, l_Q=1}^L \left(\sum_{r_1, s_1, \dots, r_Q, s_Q=1}^N k_Q(r_1, s_1, \dots, r_Q, s_Q) b_{l_1}(r_1, s_1) \dots b_{l_Q}(r_Q, s_Q) \right) \alpha_{l_1}(\mathbf{x}) \dots \alpha_{l_Q}(\mathbf{x}) \\ &+ \varepsilon \end{aligned} \tag{2.31}$$

Substituting (2.27)-(2.29) into (2.31), we obtain (2.7). \square

Proposition 2. Given $y_{Q,k}(\mathbf{x}) : \mathbb{R}^{N \times N} \rightarrow \mathbb{R}$, as in (2.3), assume that a vector space \mathcal{S} , spanned by a set of basis functions $\mathcal{B} = \{\mathbf{b}_1, \dots, \mathbf{b}_L : \mathbf{b}_l \in \mathbb{R}^{N \times N}\}$, satisfies (2.8).

Consider the set projection coefficients of the kernels in the space \mathcal{S} given in (2.27)-(2.29), and the projected kernels in the space \mathcal{S} , \hat{k}_i , obtained by using these projection coefficients in (2.14)-(2.16), Then it is true that

$$y_{Q,k}(\mathbf{x}) = y_{Q,\hat{k}}(\mathbf{x}) \quad \forall \mathbf{x} \in X, \quad (2.32)$$

Proof. From (2.3),

$$\begin{aligned} y_{Q,\hat{k}}(\mathbf{x}) &= \hat{k}_0 \\ &+ \sum_{i,j=1}^N \hat{k}_1(i,j)x(i,j) \\ &+ \dots \\ &+ \sum_{i_1,j_1,\dots,i_Q,j_Q=1}^N \hat{k}_Q(i_1,j_1,\dots,i_Q,j_Q)x(i_1,j_1)\dots x(i_Q,j_Q) + \varepsilon \end{aligned} \quad (2.33)$$

Substituting (2.14)-(2.16) into (2.33), we see that

$$\begin{aligned} y_{Q,\hat{k}}(\mathbf{x}) &= q_0 \\ &+ \sum_{i,j=1}^N \left(\sum_{l=1}^L q_1(l)b_l(i,j) \right) x(i,j) + \dots \\ &+ \sum_{i_1,j_1,\dots,i_Q,j_Q=1}^N \left(\sum_{l_1,\dots,l_Q=1}^L q_Q(l_1,\dots,l_Q)b_{l_1}(i_1,j_1)\dots b_{l_Q}(i_Q,j_Q) \right) \\ &x(i_1,j_1)\dots x(i_Q,j_Q) + \varepsilon \end{aligned} \quad (2.34)$$

Substituting (2.27)-(2.29) into (2.34), we then find that

$$\begin{aligned} y_{Q,\hat{k}}(\mathbf{x}) &= k_0 \\ &+ \sum_{i,j=1}^N \left[\sum_{l=1}^L \left(\sum_{r,s=1}^N k_1(r,s)b_l(r,s) \right) b_l(i,j) \right] x(i,j) + \dots \\ &+ \sum_{i_1,j_1,\dots,i_Q,j_Q=1}^N \left[\sum_{l_1,\dots,l_Q=1}^L \left(\sum_{r_1,s_1,\dots,r_Q,s_Q=1}^N k_Q(r_1,s_1,\dots,r_Q,s_Q)b_{l_1}(r_1,s_1)\dots b_{l_Q}(r_Q,s_Q) \right) \right. \\ &\left. b_{l_1}(i_1,j_1)\dots b_{l_Q}(i_Q,j_Q) \right] x(i_1,j_1)\dots x(i_Q,j_Q) + \varepsilon \end{aligned} \quad (2.35)$$

Interchanging the order of summations, (2.35) can be expressed as

$$\begin{aligned}
y_{Q,\hat{k}}(\mathbf{x}) &= k_0 & (2.36) \\
&+ \sum_{r,s=1}^N k_1(r,s) \left[\sum_{l=1}^L \left(\sum_{i,j=1}^N b_l(i,j)x(i,j) \right) b_l(r,s) \right] + \dots \\
&+ \sum_{r_1,s_1,\dots,r_Q,s_Q=1}^N k_Q(r_1,s_1,\dots,r_Q,s_Q) \\
&\quad \left[\sum_{l_1=1}^L \left(\sum_{i_1,j_1=1}^N b_{l_1}(i_1,j_1)x(i_1,j_1) \right) b_{l_1}(r_1,s_1) \right] \dots \\
&\quad \left[\sum_{l_Q=1}^L \left(\sum_{i_Q,j_Q=1}^N b_{l_Q}(i_Q,j_Q)x(i_Q,j_Q) \right) b_{l_Q}(r_Q,s_Q) \right] \\
&+ \varepsilon
\end{aligned}$$

Substituting (2.6) into (2.36), and using (2.5) and (2.8) we see that

$$\begin{aligned}
y_{Q,\hat{k}}(\mathbf{x}) &= k_0 \\
&+ \sum_{r,s=1}^N k_1(r,s) \left(\sum_{l=1}^L \alpha_l(\mathbf{x}) b_l(r,s) \right) + \dots \\
&+ \sum_{r_1,s_1,\dots,r_Q,s_Q=1}^N k_r(r_1,s_1,\dots,r_Q,s_Q) \left(\sum_{l_1=1}^L \alpha_{l_1}(\mathbf{x}) b_{l_1}(r_1,s_1) \right) \dots \\
&\quad \left(\sum_{l_Q=1}^L \alpha_{l_Q}(\mathbf{x}) b_{l_Q}(r_Q,s_Q) \right) + \varepsilon = y_{Q,k} \left(\sum_{l=1}^L \alpha_l(\mathbf{x}) b_l(r,s) \right) \\
&= y_{Q,k}(\Pi_S(\mathbf{x})) = y_{Q,k}(\mathbf{x})
\end{aligned}$$

□

2.6.2 True Volterra Kernels

To simplify notation, we use 1D vectors to represent input images, by concatenating the rows of their 2D representation. Then, the Volterra representation of the response is given by

$$\begin{aligned}
 y_{Q,k}(\mathbf{x}) &= k_0 + \sum_{i=1}^{N \times N} k_1(i)x(i) + \dots \\
 &+ \sum_{i_1, \dots, i_Q=1}^{N \times N} k_2(i_1, \dots, i_Q)x(i_1) \dots x(i_Q) + \varepsilon
 \end{aligned} \tag{2.37}$$

If the function describing the response model, $y_{Q,k}(\mathbf{x})$, is analytic on an open set $|\mathbf{x}| < R_0$ centered at zero with radius R_0 then at each point \mathbf{x} in that set, $y_{Q,k}(\mathbf{x})$ has the Taylor-series representation (Brown & Churchill, 1996)

$$\begin{aligned}
 y_{Q,k}(\mathbf{x}) &= y_{Q,k}(\mathbf{0}) + \sum_{i=1}^N \frac{\partial y_{Q,k}(\mathbf{0})}{\partial x_i} x(i) + \dots \\
 &+ \sum_{i_1, \dots, i_Q=1}^N \frac{1}{Q!} \frac{\partial^Q y_{Q,k}(\mathbf{0})}{\partial x_{i_1} \dots \partial x_{i_Q}} x(i_1) \dots x(i_Q) + \varepsilon
 \end{aligned} \tag{2.38}$$

By the uniqueness of Taylor series, comparing (2.37) and (2.38), we derive the true Volterra kernels

$$\begin{aligned}
 k_0 &= y_{Q,k}(\mathbf{0}) \\
 k_1(i) &= \frac{\partial y_{Q,k}(\mathbf{0})}{\partial x_i} \\
 &\dots \\
 k_Q(i_1, \dots, i_Q) &= \frac{1}{Q!} \frac{\partial^Q y_{Q,k}(\mathbf{0})}{\partial x_{i_1} \dots \partial x_{i_Q}}
 \end{aligned} \tag{2.39}$$

The functions describing the simple- and complex-cell models in (2.17) and (2.18) respectively are analytic. Hence, their analytic kernels can be calculated from (2.39).

2.6.3 Projection Pursuit Regression

The Projection Pursuit Regression (PPR) model represents the response of the system, y , to its input, $\mathbf{x} = (x_1, \dots, x_N)'$, using the non-linear transformation given in (2.19) (Friedman & Stuetzle, 1981; Friedman, 1984a). The projection part of the name indicates that the input vector \mathbf{x} is projected onto the direction-vectors $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{M_0}$. In turn, the pursuit part indicates that an optimization technique is used to find a good set of direction-vectors. The estimation of the PPR model parameters is a two-stage process. They comprise a forward stepwise procedure followed by a backward stepwise procedure. The PPR algorithm is controlled by two parameters, M and M_0 . Parameter M controls the forward procedure and parameter M_0 the backward procedure.

2.6.3.1 Forward stepwise procedure

For the forward procedure, an initial M -term model of the form given by (2.19) is constructed. First, a trial direction \mathbf{a}_1 is used to compute the values $z_{i1} = \langle \mathbf{a}_1, \mathbf{x}_i \rangle$, $i = 1, \dots, n$, where $\mathbf{x}_i = (x_{i1}, \dots, x_{iN})'$ is the i^{th} input and n is the number of inputs. Then, denoting the response of the system to input \mathbf{x}_i by y_i and $\tilde{y}_i^1 = y_i - \bar{y}$, a scatter plot of (\tilde{y}_i^1, z_{i1}) is constructed. This scatter plot is smoothed to obtain a first estimate ϕ_1 in (2.19). Next, \mathbf{a}_1 is varied to minimize the sum of squares $\sum_{i=1}^n (\tilde{y}_i^1 - \phi_1(z_{i1}))^2$, where for each \mathbf{a}_1 in the optimization procedure, a new ϕ_1 is computed using the smoothing procedure. The result of this estimation, a the pair (\mathbf{a}_1, ϕ_1) , is then standardized according to (2.19), and the corresponding value β_1 is calculated. We now have the approximation $y_i = \bar{y} + \beta_1 \phi_1(\langle \mathbf{a}_1, \mathbf{x}_i \rangle)$, $i = 1, \dots, n$. Next we treat $\tilde{y}_i^2 = y_i - \bar{y} - \beta_1 \phi_1(z_{i1})$ as the response. We then fit a second term $\beta_2 \phi_2(z_{i2})$ to this modified response in the same manner that we fitted $\beta_1 \phi_1(z_{i1})$ to \tilde{y}_i^1 . This gives the approximation $y_i = \bar{y} + \beta_1 \phi_1(\langle \mathbf{a}_1, \mathbf{x}_i \rangle) + \beta_2 \phi_2(\langle \mathbf{a}_2, \mathbf{x}_i \rangle)$, $i = 1, \dots, n$. Continuing in this fashion, we arrive at the forward stepwise estimated model

$$y_i = \bar{y} + \sum_{m=1}^M \beta_m \phi_m(z_{im}) \quad i = 1, \dots, n \quad (2.40)$$

2.6.3.2 Backward stepwise procedure

Next, models of decreasing order $m = M - 1, M - 2, \dots, M_0$ are fit in a backward stepwise manner. For each term in the model, the sum of squared residuals

$$\sum_{i=1}^n \left[y_i - \bar{y} - \sum_{l=1}^m \beta_l \phi_l(z_{il}) \right]^2 \quad (2.41)$$

is minimized with respect to β_l , \mathbf{a}_l , and ϕ_l . The initial values for these parameters are the solutions for the m most important values of the $m + 1$ terms in the previous order $m + 1$ model, where importance is measured by $|\beta_l|$.

Chapter 3

THE EXTENDED PROJECTION PURSUIT REGRESSION (ePPR)

ALGORITHM

A central goal of systems Neuroscience is to characterize the transformation of sensory input to spiking output in single neurons. This problem is complicated by the large dimensionality of the inputs. To cope with this problem, previous methods have estimated simplified versions of a generic linear-nonlinear (LN) model and required, in most cases, stimuli with constrained statistics. Here we develop the extended Projection Pursuit Regression (ePPR) algorithm that allows the estimation of all the parameters, in space and time, of a generic LN model using arbitrary stimuli. We first prove that ePPR models can uniformly approximate, to an arbitrary degree of precision, any continuous function. To test this generality empirically, we use ePPR to recover the parameters of models of cortical cells that cannot be represented exactly with an ePPR model. Next we evaluate ePPR with physiological data from primary visual cortex and show that it can characterize both simple and complex cells from their responses to both natural and random stimuli. For both simulated and physiological data, we show that ePPR compares favorably to spike-triggered and information-theoretic techniques for the estimation of the filters of LN models. To our knowledge ePPR is the first algorithm to estimate, from natural and random stimuli, a generic LN model that is spatio-temporal, uses two-dimensional images as inputs, and whose linear component contains more than one filter. Thus, ePPR may

be a good standard model to unify the characterization of all visual cells in early stages of the visual system.

3.1 Introduction

One of the most important problems in Neuroscience is to characterize functionally how sensory neurons transform their input to spiking output. Two central issues for these characterizations are the stimuli used to probe a cell and the model used to represent it. Regarding the stimuli, traditionally sensory neurons have been studied with small sets of simple stimuli, specifically designed to probe certain aspects of their response properties (e.g., Liberman, 1982; Hartline, 1940; Kuffler, 1953; Barlow, 1953), or with large sets of random stimuli (e.g., P. Marmarelis & Marmarelis, 1978). However, recent work (Theunissen, Sen, & Doupe, 2000; David et al., 2004; Felsen et al., 2005; Wooley et al., 2006; Sharpee et al., 2006, 2008) has shown that observable properties of sensory cells depend on the statistical properties of the stimuli used to probe them. Therefore, to understand how sensory cells operate in natural conditions, it is important to characterize them from their responses to natural stimuli (Felsen & Dan, 2005). Regarding the model, cells of different classes are normally characterized with different parametric models. Based on prior evidence, a model that contains the relevant structure for a class of cells is proposed, and parameters of this model are fitted to experimental data from a cell of this class. A problem with this approach is that the hypothesized model structure may not be correct, making the interpretation of the fitted parameters questionable. An alternative is to use generic models that can well approximate cells from a large set of classes. Here we address both of these issues by developing a method to estimate a generic model that can characterize responses of many classes of cells to arbitrary, including natural, stimuli. We focus the description and evaluation of this method on the visual system, though the method is applicable to other sensory modalities.

Using natural images to characterize visual cells is non-trivial. Natural images are complex (Simoncelli & Olshausen, 2001), so the number of descriptors needed to represent them is large, and a generic model would need a very large number of parameters to characterize responses of visual cells to natural stimuli. Due to the curse of dimensionality (Bellman, 1961) the amount of data required to estimate the parameters of a model grows exponentially with the number of model parameters. Therefore, a prohibitively large amount of data –unattainable in standard physiology experiments– would be required to estimate the parameters of generic models of visual cells using natural stimuli as inputs.

A common strategy to overcome this problem is to assume that the cell response follows a LN model. A generic version of this model is shown in Figure 3.1. The response of the model at time bin i is assumed to depend on the image presented at time bin i , plus the images presented on the previous D time bins. At each delay d , the model contains M_d filters. To generate its response, the model computes the dot product between the input image at delay d and each of the M_d filters at that delay, generating scalars $g_{1,d}, \dots, g_{M_d,d}$. Then the scalars at all delays, $\{g_{m,d}, 0 \leq d \leq D, 1 \leq m \leq M_d\}$, are used as inputs to a nonlinear function N , that predicts the cell’s spike rate at time bin i ¹.

However, even this generic LN model has a large number of parameters. For instance, for images of size 32×32 pixels, a model with $M_d = 3$ for all d and $D = 4$ will contain $32 \times 32 \times 3 \times (1 + 4) = 15,360$ filter parameters, as well as additional parameters needed to describe the nonlinear function N . To avoid problems associated with estimating such a large number of parameters, previous techniques (Chichilnisky, 2001; Sharpee et al., 2004; Touryan et al., 2005; Rapela et al., 2006) have estimated simplified versions of this model. STA can only estimate one filter per delay, i.e., $M_d = 1$ for all d (Figure 3.1, red box), and requires a radially symmetric distribution of the input images (Chichilnisky, 2001; Paninski, 2003). STC allows the estimation of multiple filters, but has more stringent

¹A more standard definition of a LN model would use spatio-temporal filters, concatenating the i^{th} filters at all delays to give the i^{th} spatio-temporal filter. But our definition is more convenient to introduce of the extended Projection Pursuit Regression model below, while both definitions are equivalent.

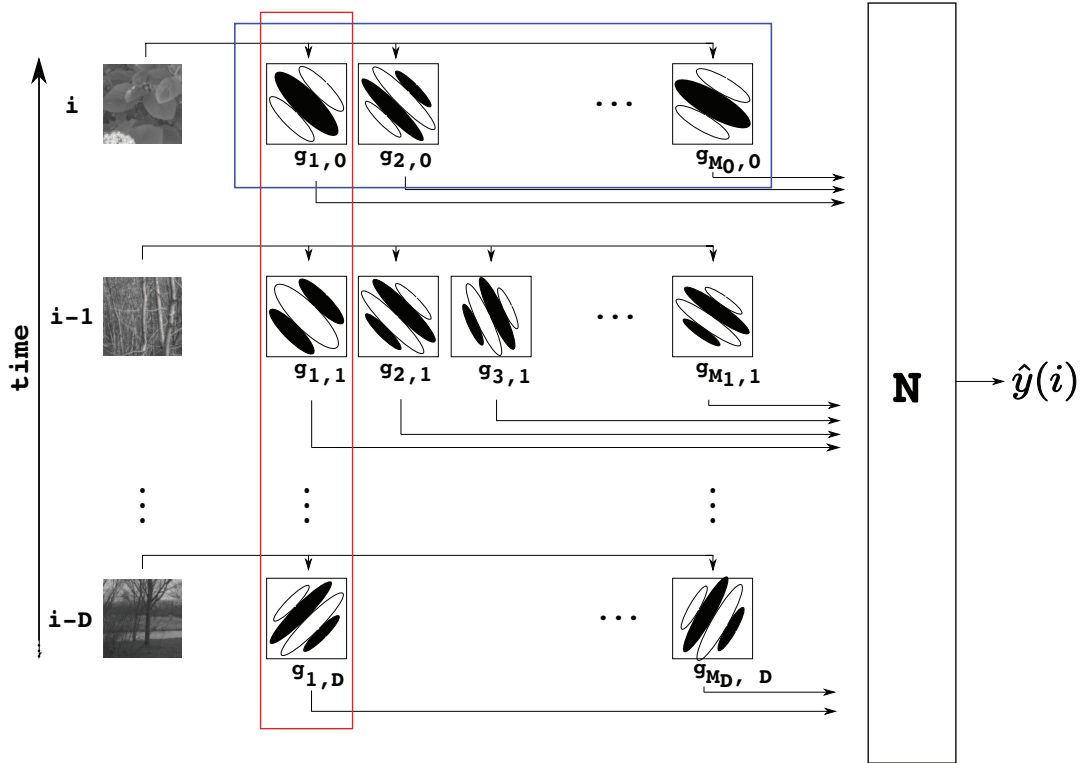


Figure 3.1: A generic LN model. The prediction of the model at time bin i , $\hat{y}(i)$, depends on the image presented at time bin i , plus the images presented at the previous D time bins. At each delay d , the model contains M_d filters. To generate its output, the model projects the input image at delay d on the M_d filters at that delay, generating scalars $g_{1,d}, \dots, g_{M_d,d}$. Then, the scalars at all delays, $\{g_{m,d}, 0 \leq d \leq D, 1 \leq m \leq M_d\}$, are used as inputs to a nonlinear function N , that predicts the cell's spike rate a time bin i . Because this model has too many parameters, previous methods have estimated simplified versions of this model. The red box surrounds the filters of the models estimated by Chichilnisky et al. (2001) and by Sharpee et al. (2006, 2008). The blue box surrounds the filters of the spatial models estimated by Touryan et al. (2005) and by Rapela et al. (2006).

convergence conditions than STA, in that the distribution of the input images needs to be Gaussian (Paninski, 2003). Thus, neither STA or STC work with natural stimuli. Touryan et al. (2005) and Rapela et al (2006) proposed techniques that can use natural stimuli, but neglect the temporal dimension of the inputs, i.e., $D = 0$ (Figure 3.1, blue box). The method introduced by Sharpee et al (2004) can estimate spatio-temporal models from responses of visual cells to natural images. But with physiological data this methods has only been used to estimate models with one filter per delay, i.e., $M_d = 1$ for all d (Sharpee et al., 2006, 2008, Figure 3.1, red box). Using filters estimated by Spike Triggered Covariance (de Ruyter van Steveninck & Bialek, 1988) and a parametric model for the nonlinear function, Rust et al (2005) estimated all parameters of the generic LN model in Figure 3.1, but at the cost of restricting the stimulus ensemble to vary along a single spatial dimension (i.e., binary random bars), and constraining the statistics of the inputs to be Gaussian white noise.

An alternative to simplifying the general LN model in Figure 3.1, or constraining the statistics of its inputs, is to use an efficient optimization strategy to enable the estimation of all its parameters using arbitrary inputs. The Projection Pursuit Regression algorithm (PPR, Friedman & Stuetzle, 1981) provides one such strategy. By decomposing a high-dimensional estimation problem into a sequence of lower dimensional ones, PPR is one of the few multivariate methods able to bypass the curse of dimensionality (Huber, 1985). In Rapela et al. (2006) we showed that PPR compared favorably with previous methods for the spatial characterization of visual cells, i.e., characterizations as a function of only one image presented to the cell before its response. However, responses of a visual cells are not a spatial, but spatio-temporal, i.e., responses of visual cells depend on several images presented to the cell before its response. For spatio-temporal characterizations of visual cells, the efficient optimization strategy of PPR is not sufficient to escape the curse of dimensionality. Here, we introduce the extended Projection Pursuit Regression (ePPR) algorithm, which extends the PPR algorithm to allow the estimation of all the parameters of the model in Figure 3.1 using natural images. In addition, the ePPR

estimation algorithm is designed to be robust to correlations in natural images, and the ePPR estimation problem is regularized to allow estimations using images of large size as inputs.

The ePPR model is very general. Below we prove that it can uniformly approximate, to an arbitrary degree of precision, any continuous function with inputs in the unit cube. To test this generality empirically, we use ePPR to recover the parameters of a LN model of a complex cell with divisive normalization, and a linear-nonlinear-linear (LNL) extension of it, neither of which can be represented exactly with an ePPR model. We then evaluate if, despite the mismatch, ePPR provides good approximations of the LN and LNL models. Next we test this generality with physiological data, by using ePPR to characterize cortical complex and simple cells. ePPR models can be estimated with stimuli with arbitrary statistics. To validate this, for the simulated and cortical cells studied in this article, we compare ePPR models estimated from their responses to natural and random data.

The rest of the article is organized as follows. Section 3.2 summarizes the ePPR algorithm (Details are provided in Appendix 3.8.2). Section 3.3 presents the results of the application of ePPR to recover the parameters of the simulated cells. Next we use ePPR to characterize a complex cell (Section 3.4) and simple cell (Section 3.5). We discuss advantages and disadvantages of ePPR, and draw final conclusions, in Section 3.6. Supplementary information is provided in appendices.

3.2 Extended Projection Pursuit Regression

In this section we introduce the ePPR algorithm. Because ePPR extends PPR, we start by summarizing the PPR algorithm in Section 3.2.1. Then Sections 3.2.2-3.2.4 describe the extensions introduced into PPR. Detailed algorithmic descriptions of PPR and ePPR are given in Appendix 3.8.1 and Appendix 3.8.2, respectively.

3.2.1 Projection Pursuit Regression

The PPR model is show in Equation 3.1. For an input $x_i \in \mathbb{R}^n$ the output of the PPR model $\hat{y}(i) \in \mathbb{R}$ is given by the mean response \bar{y} plus M_0 terms of the form $\beta_m \phi_m(\alpha_m^T x_i)$. In these terms, $\beta_m \in \mathbb{R}$ is an importance coefficient, and ϕ_m is a uni-dimensional nonlinear function acting on the dot product between the unit-norm direction $\alpha_m \in \mathbb{R}^n$ and x_i .

$$\hat{y}(i) = \bar{y} + \sum_{m=1}^{M_0} \beta_m \phi_m(\alpha_m^T x_i) \quad (3.1)$$

with $\|\alpha_m\|_2 = 1$, $\frac{1}{n} \sum_{i=1}^n \phi_m(\alpha_m^T x_i) = 0$, and $\frac{1}{n} \sum_{i=1}^n \phi_m^2(\alpha_m^T x_i) = 1$

The algorithm used to estimate the parameters of the PPR model finds the prediction function \hat{y} minimizing the sum of squared errors (SSE) with the response function y (Equation 3.2). To escape from the curse of dimensionality, it estimates one term of Equation 3.1 at a time, as follows: Having determined the functions $\phi_1, \phi_2, \dots, \phi_{m-1}$ and the unit vectors $\alpha_1, \alpha_2, \dots, \alpha_{m-1}$, the estimation algorithm chooses a unit vector α_m and a function ϕ_m that minimize the SSE in Equation 3.3. To avoid local minima in this greedy procedure, PPR uses a backward stepwise procedure.

$$SSE = \sum_{i=1}^N (y(i) - \hat{y}(i))^2 \quad (3.2)$$

$$SSE = \sum_{i=1}^N (r_m(i) - \phi_m(\alpha_m^T x_i))^2, \text{ where } r_m(i) = y(i) - \sum_{j=1}^{m-1} \beta_j \phi_j(\alpha_j^T x_i) \quad (3.3)$$

Approximation theory results for PPR

What types of functions can be well approximated by PPR models?

PPR models can represent exactly a large class of functions. For instance, any multi-dimensional polynomial can be represented exactly by a PPR model (Proposition 1 in Appendix 3.8.3). Although not all functions admit an exact PPR representation, any continuous function, with inputs in $[0, 1]^p$, can be uniformly approximated, to an arbitrary degree of precision, by a PPR model. This follows from (1) the Stone-Weierstrass theorem (Rudin, 1976) which implies that polynomials are dense on $[0, 1]^p$, and (2) the previous result that polynomials can be represented exactly by PPR models.

It is conceivable that a given function could be well approximated by a PPR model but that the greedy PPR estimation algorithm does not converge to the optimal approximation. So, the following question becomes relevant.

Under what conditions will the PPR estimation algorithm converge to its best approximation?

For inputs x , samples from a probability distribution P uniform on the unit ball or multivariate Gaussian, Donoho et al. (1985) announced a proof of strong convergence, i.e., $r_m \rightarrow 0$ in the norm of $L_2(P)$. In addition, Jones (1987) proved strong convergence for general P , when the nonlinear functions are given by the conditional expectations (Equation 3.4) and the estimated projection directions are uniformly close to the optimal greedy projection directions (Equation 3.5).

$$\phi_m(z) = E(r_m(X) | \alpha_m^T X = z) \tag{3.4}$$

$$E(\phi_m(\alpha_m^T X))^2 > \rho \sup_{b^T b=1} E(\phi_m(b^T X))^2, \rho \text{ fixed}, 0 < \rho < 1. \tag{3.5}$$

However, with empirical data these conditions cannot be verified and one cannot know if PPR has converged to the optimal solution.

3.2.2 Spatio-Temporal models

In Rapela et al. (2006), we estimated spatial PPR models of visual cells. The responses of these models depended on a single image presented to the cell before its response. However, we would like to estimate spatio-temporal PPR models, i.e., models whose response depends on several past images, as is known to be the case for most visual cells.

To predict the cell response at time bin i , in Rapela et al. (2006) we used as input to PPR the vector representation of the image presented at time bin i . Calling $I_i \in \mathbb{R}^{p \times p}$ the image presented at time i , and $\text{vec} : \mathbb{R}^{p \times p} \rightarrow \mathbb{R}^{p^2}$ the operation that transforms an image into its vector representation, the input to PPR in Rapela et al. (2006) was $x_i = \text{vec}(I_i) \in \mathbb{R}^{p^2}$.

A first type of spatio-temporal model uses the spatial PPR model in Equation 3.1, but taking as input the concatenation of the images presented at time bins $\{i, \dots, i-D\}$, i.e., $x_i = [\text{vec}(I_i)^T, \dots, \text{vec}(I_{i-D})^T]^T \in \mathbb{R}^{(D+1)p^2}$. As PPR models, spatio-temporal models of this type, can uniformly approximate, to an arbitrary degree of precision, any continuous functions with inputs in the unit cube. A limitation of this type of model is that the dimensionality of the inputs grows in proportion to the memory D of the model, which complicates the estimation of the filters α_m in Equation 3.1. Also, to build these models, the memory D must be determined in advance.

A second type of spatio-temporal model is obtained by adding to Equation 3.1 extra terms operating on images at different delays, as shown in Equation 3.6. The estimation algorithm for this type of model requires only simple modifications to the PPR estimation algorithm (Appendix 3.8.2). An advantage of this type of model is that the dimensionality of the inputs does not grow with the memory D of the model, simplifying the estimation of the filters. In addition, the estimation algorithm learns, in a single run, the optimal memory, D , required to characterize a cell. A disadvantage is that Equation 3.6 is not the most general spatio-temporal extension of the PPR model. In particular, pixels of images occurring at different delays are contained in different terms in Equation 3.6, and

²In the vector representation, the columns of an image are concatenated together to form a long vector.

different terms are combined linearly. Therefore, nonlinear interactions between pixels of images at different delays cannot influence the predictions of this type of model.

$$\hat{y}(i) = \bar{y} + \sum_{d=0}^D \sum_{m=1}^{M_d} \beta_{m,d} \phi_{m,d}(\alpha_{m,d}^T x_{i-d}) \quad (3.6)$$

with $\frac{1}{n} \sum_{i=1}^n \phi_{m,d}(\alpha_{m,d}^T x_{i-d}) = 0$, and $\frac{1}{n} \sum_{i=1}^n \phi_{m,d}^2(\alpha_{m,d}^T x_{i-d}) = 1$

We call models of the first type ePPR models with time interaction, and models of the second type ePPR models without time interaction. Selecting between these types of model is a tradeoff between model generality and optimization feasibility. When the memory of a cell is short, and/or the response sampling rate is small, and/or there is enough physiological data to make possible the estimation of the larger spatio-temporal filters, then models with time interaction are preferable. One possibility to determine the memory of the cell is to estimate an ePPR model without time interactions. Then, if the memory, D , of this model is sufficiently short, or the amount of data is sufficiently large, a more general ePPR model with time interactions can be estimated. In what follows we will refer to Equation 3.6 as the ePPR model, with the caveat that for models with time interaction several images are concatenated into a single input, as indicated above, and the memory of the model is set to $D = 0$.

3.2.3 Robustness to correlations in natural images

With simulated data we observed that the filters of PPR models estimated from natural data were significantly worse than those estimated from random data. As noted in Appendix 3.8.1, a key step in PPR is the solution of a nonlinear least-squares problem for estimating a projection direction α_m (minimization of the SSE in Equation 3.14). In PPR, this problem is solved using the Gauss-Newton method (Nocedal & Wright, 2006), which, in turn, requires the solution of a linear system of equations $Ax = b$, Equation 3.17. The Gauss-Newton algorithm is guaranteed to converge if the eigenvalues of A are bounded

away from zero (Nocedal & Wright, 2006). This is the case for the matrix A constructed from random data, but not for the one constructed from natural data, which cause A to have zero eigenvalues. Thus, for natural data, the Gauss-Newton is not guaranteed to converge. To overcome this problem, we replaced the Gauss Newton by a Trust Region method (Nocedal & Wright, 2006). The latter method is guaranteed to converge under very general conditions, which do not require non-zero eigenvalues (Nocedal & Wright, 2006).

3.2.4 Smooth prior for large-dimensional filters

Responses of visual cells contain high levels of noise. Then, estimating ePPR parameters by minimizing Equation 3.2 leads to estimates that overfit the noise, i.e., narrowly focusing on the training error is likely to return estimates that describe the training set well, but perform poorly in predicting responses to novel data. A common strategy to overcome this problem is to penalize the estimated function, \hat{y} , based on some a priori measure of how likely \hat{y} is to overfit noise. In ePPR, \hat{y} is penalized for containing non-smooth projection directions $\alpha_{m,d}$. The ePPR objective function, Equation 3.7, contains a penalty term $\lambda \sum_{d=0}^D \sum_{m=1}^{M_d} \|L\alpha_{m,d}\|^2$. In this term, L is a smoothing operator such that $\|L\alpha_{m,d}\|^2$ will be large when $\alpha_{m,d}$ is non-smooth. In turn, the regularization parameter λ controls the tradeoff between fitting the responses accurately and estimating smooth filters $\alpha_{m,d}$. For the results shown below, we chose L to be a 3×3 Laplacian operator. To select the parameters λ , we used the procedure described in Appendix 3.8.2.2.

$$J = \sum_{i=1}^N (y_i - \hat{y}(x_i))^2 + \lambda \sum_{d=0}^D \sum_{m=1}^{M_d} \|L\alpha_{m,d}\|^2 \quad (3.7)$$

In summary, ePPR fits the spatio-temporal model in Equation 3.6 by optimizing the criterion in Equation 3.7, using a Trust Region method that is robust to correlations

in natural images. An implementation of the ePPR algorithm can be downloaded from <http://vpl.usc.edu/projects/ePPR/>.

3.3 Simulated cell

Here we evaluate ePPR for the characterization of a simulated complex cell. For the simulations we used a simplified version of a divisive gain control model, as have been used to describe nonlinear properties of neurons in primary visual cortex (Heeger, 1992). The mean response of the simulated cell at time i is given by $\bar{y}(i)$ in Equation 3.8, where $[x(i)^T, x(i-1)^T, x(i-2)^T]$ stands for the (transpose of the) concatenation of the images at time i , $i-1$, and $i-2$. The noisy response is a Poisson random variable with this mean. The spatio-temporal filters, f_1, f_2, f_3 , used in this model are shown in Figure 3.2a. The filters f_1 and f_2 are facilitatory, because larger dot products with these filters produce stronger mean responses, while the filter f_3 is suppressive, because larger dot products with this filter produce weaker mean responses. The parameter γ controls of the mean of the responses and, because the noise was Poisson, it also controls the level of noise in the responses. To study the effect of noise in the estimated models, we varied γ to generate 3 sets of 24,000 responses with different levels of noise. For all three sets we fixed the inhibitory constant ω so that the mean of the denominator in Equation 3.8, for all simulated responses, was 4.26, i.e., on average the divisive normalization reduced the unnormalized cell response by more than four times. We study the effects of varying the amount of inhibition in Appendix 3.8.5. Further details on the simulation procedure are given in Section 3.7.3. Note that, due to the divisive normalization, Equation 3.8 cannot be represented exactly by an ePPR model. Thus, this model tests the generalization of ePPR. To further test this generalization, in Section 3.3.3 we use ePPR to characterize a LNL extension of this complex cell model.

$$\bar{y}(i) = \gamma \frac{([x(i)^T, x(i-1)^T, x(i-2)^T]f_1)^2 + ([x(i)^T, x(i-1)^T, x(i-2)^T]f_2)^2}{1 + \omega ([x(i)^T, x(i-1)^T, x(i-2)^T]f_3)^2} \quad (3.8)$$

The ePPR estimation procedure makes no assumption about the statistical properties of the stimuli used to estimate its model parameters. To validate this, we estimated ePPR models from simulated responses to natural and random ensembles. For the real complex and simple cells the natural ensemble (quasi-natural sequence ensemble, Section 3.7.2) approximates the spatial statistics of natural movies, but is temporally uncorrelated. To mimic this, for the simulated cell we used a reshuffled natural movie as the natural ensemble (natural sequence ensemble, Section 3.7.2). But, to assess the influence of temporal correlations on ePPR estimates, we also estimated ePPR models from responses to the unshuffled natural movie (Appendix 3.8.6).

Nonlinear interactions between pixels of images at different delays are relevant to the simulated model in Equation 3.8. Accordingly, predictions from ePPR models with time interaction were significantly better than those of ePPR models without time interaction (Figure 3.10c). In this section we show the parameters of ePPR models with time interaction, and in Figures 3.10a and 3.10b (Appendix 3.8.4) we show the parameters of a model without time interaction.

All models in this section were estimated using sets of 20,000 responses, and their predictive power was evaluated using a disjoint set of 4,000 responses. The estimation of the filters and nonlinear function in ePPR is entirely nonparametric. The conjunction of this non-parametric estimation with the very large amount of noise in the simulated responses could lead to estimates with large variability. To study this variability, for each level of noise in the simulated responses, we estimated five models from distinct resampled subsets of the training data set (see Section 3.7.4). Figures 3.2d, 3.2g, and 3.2h quantify the variability in the estimated parameters. For simplicity, Figures 3.2b, 3.2c, 3.2e, and

3.2f, 3.3a, and 3.3b show parameters of example models (estimated from responses with intermediate amount of noise).

3.3.1 Data from natural images

Estimated filters

Figure 3.2b shows contour plots of the filters from the example ePPR model estimated using natural data with intermediate amount of noise. The title of each contour plot is the corresponding β coefficient. Qualitatively the estimated filters look very similar to the true filters of the simulated model (Figure 3.2a). To quantify this similarity, we computed the principal angles between true and ePPR filter spaces (Section 3.7.5). These principal angles show how much of the three dimensional structure of the true filter space is well approximated by the estimated filter space. If the three principal angles are relatively small, then the estimated filter space well approximates the true filter space along its three dimensions. But, if only the first n principal angle are small, then the estimated filter space is a good approximation of the true filter space only along n dimensions. The orange curve in Figure 3.2d plots the averaged principal angles between the true and estimated filter spaces. The three principal angles are relatively small, showing that the filter space estimated by ePPR is a good approximation of the true filter space along its three dimensions. Moreover, the size of the error bars is small indicating a small variability in the estimated filters.

For comparison, we also estimated 4 sets of Maximally Informative Dimensions filters (MID; Sharpee et al., 2004, Section 3.7.9) from 4 different jackknifed subsets of the training dataset. Figure 3.3a shows an example set of MID filters. These filters are plotted from left to right in their estimation order. As in all our MID estimates for the simulated cell, the first MID filter was a good estimate, one of the other two filters was a mediocre estimate (third filter in Figure 3.3a), and the remaining filter was a poor estimate (second filter in Figure 3.3a). In addition, MID did not recovered the inhibitory filter in any of 4 sets of estimated filters. The red curve in Figure 3.2d plots the average principal

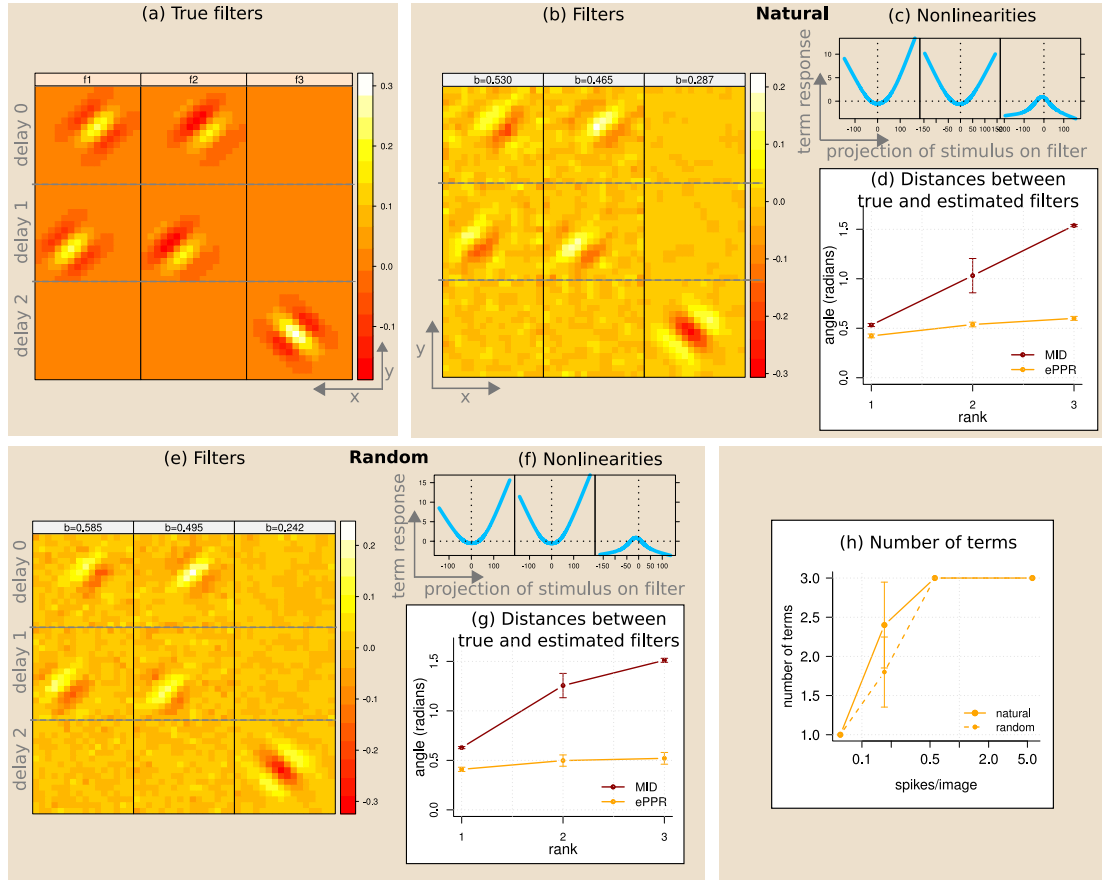


Figure 3.2: Simulated cell: ePPR models. (a): filters of the simulated model (Equation 3.8). (b,c): filters (b) and nonlinear functions (c) of the example model estimated from natural data. The titles in (b) are the corresponding β coefficients. (d): principal angles between the true filters and those of models estimated from fitting subsets of the data with intermediate level of noise. (e-g): as (b-d) but for models estimated from random data. (h) average number of terms in ePPR models estimated from natural and random data. For both example models, the estimated filters are similar to the true filters, and the nonlinear functions correctly indicate the facilitatory/suppressive nature of the corresponding filters.

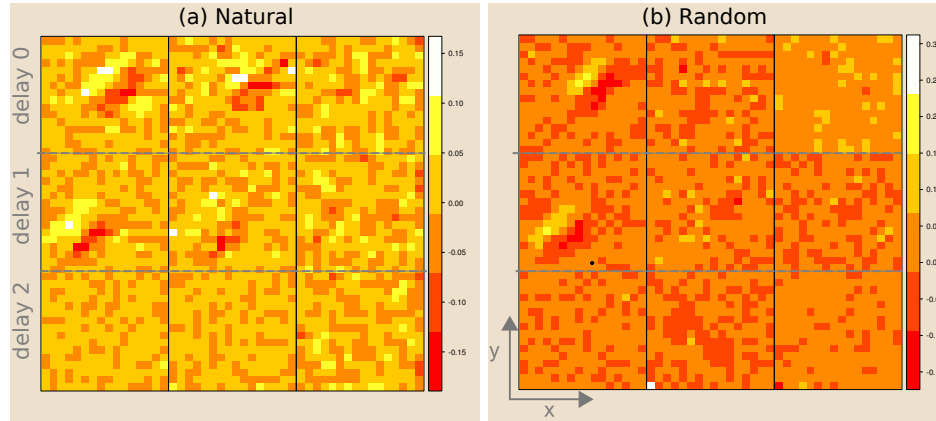


Figure 3.3: Simulated cell: example set of MID filters estimated from natural data (a) and random data (b). The filters are sorted from left to right in their estimation order. As in the other four sets of MID filters, the first filter is a good estimate, one of the other two filters (the third filter in the example) is an mediocre estimate, the remaining filter (the second filter in the example) is a poor estimate, and the suppressive filter is missing. MID well approximated the true filter space only along one dimension (see red curves in Figures 3.2d and 3.2g).

angles between the true and MID filter spaces. The small first principal angle shows that the MID filter space well approximated the true filter space along one dimension. But the larger second and third principal angles indicate that the MID approximations along more than one dimension were poor.

Estimated nonlinear functions

Figure 3.2c shows the nonlinear functions of the ePPR model estimated from natural data. These functions map the dot product of an image and the corresponding filter onto the contribution of the term to the prediction of the model. Points with large magnitude in these nonlinear functions are either all positive or all negative. If they are all positive, the filter associated with the nonlinear function will facilitate the cell response, and if they are all negative the filter will suppress the cell response. Points with large magnitude in the two leftmost nonlinear functions are all positive, while those in the rightmost nonlinear function are all negative. Thus, the two leftmost filters in

Figure 3.2b are facilitatory, while the rightmost filter is suppressive. Since this agrees with the construction of the model cell, the estimated nonlinear functions correctly recovered the facilitatory/suppressive nature of the associated filters.

Predictive power

We compared the predictive power of filters estimated from different methods: ePPR; MID; normalized spike triggered covariance (nSTC; Touryan et al., 2005, Section 3.7.10), as has been previously employed to characterize complex cells (Touryan et al., 2002; Felsen et al., 2005); and PPR (Appendix 3.8.1)³. Because some of the above methods do not provide a predictive model, and to make the comparison of the predictive power of the filters independent of the predictive model used by each method, we used a second-order multi-dimensional polynomial (Section 3.7.12) as the predictive model for all the methods. For each number of spikes per image, or noise level, in the simulated responses, and for each type of filter (ePPR, MID, PPR, or nSTC) we constructed a second-order multi-dimensional polynomial, use it to predict responses to the 8 testing subsets (Section 3.7.4), and computed the Pearson correlation coefficient between each of these predictions and the simulated cell responses. Figure 3.4a, plots the mean of these correlation coefficients, with error bars of size one standard deviation, as a function of the or noise level in the simulated responses. The orange, red, blue, and cyan curves correspond to ePPR, MID, PPR and nSTC filters, respectively. The black curve is an upper bound on the correlation coefficient that any model can achieve (Section 3.7.7). Light red asterisks indicate that correlation coefficients for ePPR filters were significantly greater than those for MID filters (Wilcoxon signed-rank test, $p < 0.01$, Section 3.7.13). The polynomial models constructed with spatio-temporal filters, ePPR and MID, predicted responses substantially better than those constructed with spatial filters, PPR and nSTC. In addition, at all noise

³PPR and nSTC, as implemented here (Section 3.7.10), can only estimate spatial filters, i.e., filters that operate on a single image presented to the cell prior to its response. For these methods, we estimate filters operating on the image presented to the cell at the same time bin as the response. This choice optimized the models predictions.

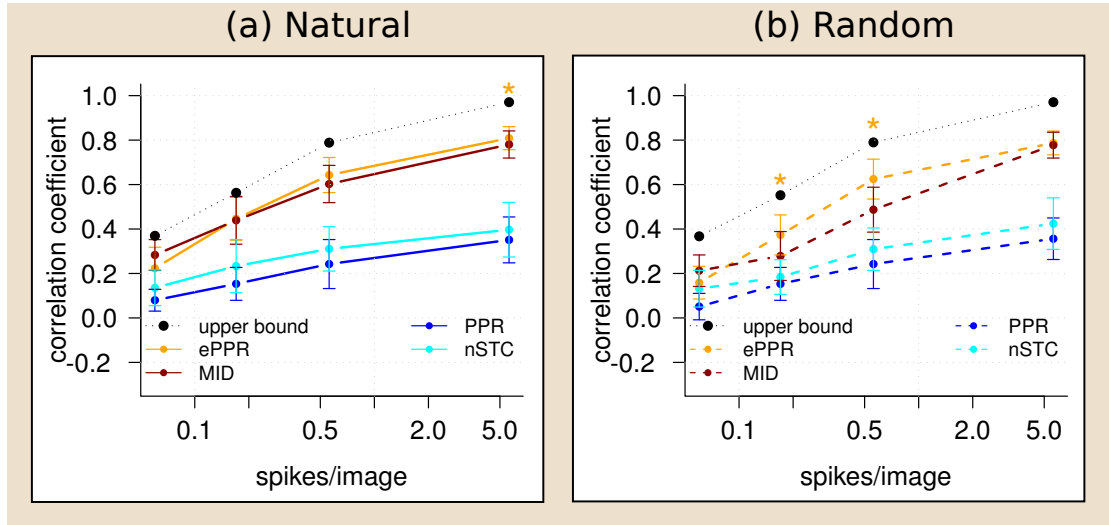


Figure 3.4: Simulated cell: predictive power of filters estimated by different methods. Pearson correlation coefficient between polynomial models predictions and simulated cell responses, as a function of the mean number of spikes per image, or noise level, in the simulated responses. The orange, red, blue, and cyan curves correspond to polynomials constructed with ePPR, MID, PPR, and nSTC filters, respectively. Black curves: upper bound on the correlation coefficients. (a): predictions with natural stimuli. (b): predictions with random stimuli. Light red asterisks mark number of spikes/image at which correlation coefficients for ePPR models are significantly greater than those for MID models. For both natural and random data, the polynomial models constructed with spatio-temporal filters, ePPR and MID, predict substantially better than those constructed with spatial filters, PPR and nSTC. Also, ePPR filters yielded significantly better predictions than MID filters for all conditions, excluding random data and intermediate amount of noise (0.56 spikes/sec).

levels, polynomial models constructed with ePPR filters predicted responses significantly better than those constructed with MID filters.

3.3.2 Data from random images

As for natural data, the filters of the example model estimated from random data look similar to the true filters of the simulated model (Figures 3.2a and 3.2e). This similarity holds for the five ePPR models estimated from the fitting subsets of the training dataset with 20,000 responses (orange curve in Figure 3.2g). As for natural data, the MID filter space well approximated the true filter space along one dimension, but the

approximation along more than one dimension was poor (Figure 3.3b and red curve in Figure 3.2g). Also, the nonlinear functions estimated from random data correctly indicated the excitatory/inhibitory nature of the corresponding filters (Figure 3.2f). Furthermore, second-order multi-dimensional polynomials constructed with ePPR or MID predicted responses substantially better than those constructed with PPR or STC filters (Figure 3.4b). In addition, polynomial models constructed with ePPR filters predicted responses significantly better than those constructed with MID filters at low and high noise levels (Wilcoxon signed-rank test, $p < 0.01$ for 0.17 and 5.62 spikes/sec).

Figure 3.2h plots the average number of terms in ePPR models as a function of the number of spikes/image, or noise level, in the simulated responses. At the intermediate (0.56 spikes/image) and low noise levels (5.62 spikes/image), all ePPR models estimated from natural or random data contained good approximations to the 3 filters of the simulated cell, and at the high noise level (0.17 spikes/image) some models missed one or two of these filters. We note that for every noise level ePPR recovered, statistically, the same number of terms from natural and random data, which is relevant for interpreting the models estimated from real cells (see below).

Due to the divisive inhibition, the simulated complex cell model cannot be represented exactly by an ePPR model. Nevertheless, ePPR provided very good approximations. These approximations remained good for responses simulated with other amounts of normalizations (Appendix 3.8.5). In addition, the simulated model in Equation 3.8 contains nonlinear interactions between pixels of images at different delays. As discussed in Section 3.2.2, these interactions cannot be accounted by ePPR models without time interactions. Nevertheless, these models provided good approximations to the true parameters of the simulated model (Figure 3.10 in Appendix 3.8.4). This shows that ePPR has good generalization properties.

3.3.3 Linear-nonlinear-linear model

The linear-nonlinear-linear model (LNL, Korenberg & Hunter, 1986) is a block structured model that found early applications in the auditory (Weiss, 1966) and visual (Weiss, 1966; Spekreijse, 1969; Spekreijse & Oosting, 1970; Spekreijse & Reits, 1982) systems. To further test the generality of ePPR, we used it to characterize a LNL extension of the previous complex cell LN model (Equation 3.8). The mean responses of the LNL model were generated by filtering the mean responses of the simulated LN model with intermediate amount of noise (Section 3.7.3) with a linear-phase lowpass filter with a cutoff frequency of $\pi/2$ cycles per sample, and with a length of 15 samples. Responses of the LNL model were Poisson random variables with these means. To illustrate the differences between the mean responses of the LN and LNL models, Figure 3.5a plots their power spectrum.

The forward ePPR model (Appendix 3.8.2) estimated from responses of the LNL model was configured to contain $n[i]$ terms at delay i with $n = [1, 1, 1, 1, 1, 1, 2, 4, 4, 4, 2, 1, 1, 1, 1]$. That is, the forward model contained terms at 15 delays. But the ePPR estimation procedure discarded the irrelevant terms, so that the final ePPR estimated model contained only terms at delays 7, 8 and 9 (Figures 3.5b, and 3.5c). Considering that the lowpass filter after the nonlinear function in the LNL model introduces a delay of 7 samples, we see that ePPR has learned the correct model structure for this LNL model.

Comparing the filters of the ePPR model without time interaction estimated from responses of the LNL model (Figure 3.5b) with those estimated from responses of the LN model (Figure 3.10a), we see that the linear filter following the nonlinearity in the LNL model has a detrimental effect on the quality of the estimated ePPR filters. For example, the true filters at delay 0 are positioned in the top-left quadrant (Figure 3.2a), but the most important estimated filter at delay 0 (Figure 3.5b, top-left panel) is incorrectly positioned at the center. Still, these filters capture important features of the true filters, like their Gabor shape, orientation, and spatial frequency. Also, the nonlinear functions

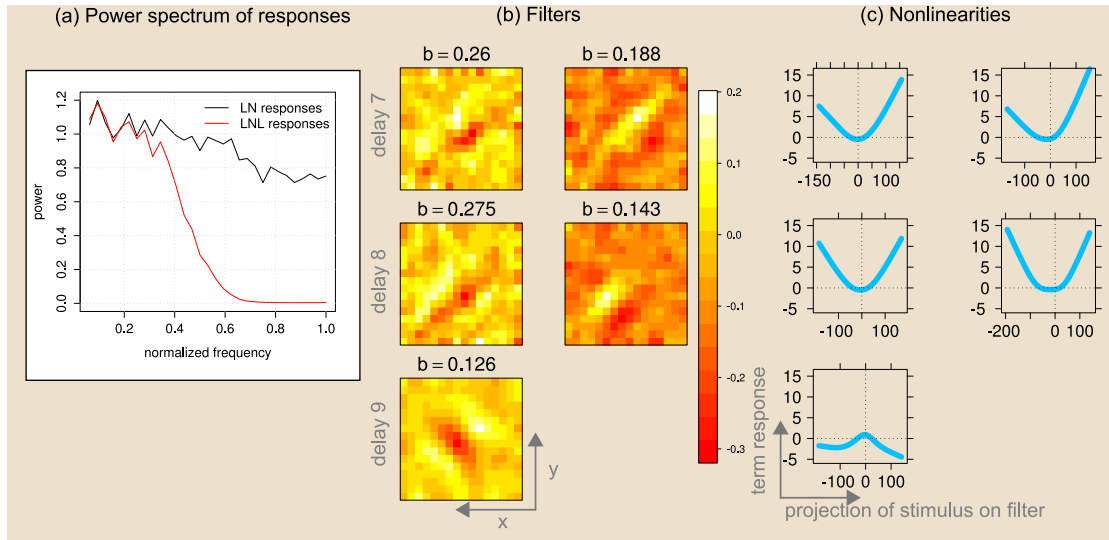


Figure 3.5: LNL model: (a): power spectrum of the responses of the LN (black curve) and LNL (red curve) models. The lowpass filter had a large effect on the responses of the LN model. (b,c): estimated filters (b) and nonlinear functions (c) at delays 7, 8, and 9. The ePPR estimation procedure discarded the irrelevant terms from the forward model and learned the correct model structure for the simulated LNL model, with two facilitatory terms at delay 7, two facilitatory terms at delay 8, and one suppressive term at delay 9. Although the filters for the LNL model are worse estimates than those for the LN model (Figure 3.10a), the former filters still are reasonably good approximations of the true filters.

have proper quadratic shapes, and correctly indicate the facilitatory or suppressive nature of the corresponding filters (Figure 3.5c).

In summary, we have shown that ePPR successfully characterized a simulated complex cell from responses to natural and random stimuli. ePPR recovered all the underlying filters and nonlinear functions from the simulated cell, including those that were suppressive. In addition, predictions from ePPR filters were superior to predictions from filters estimated by previous methods. The generality of ePPR was demonstrated by using it to characterize a simulated complex cell with divisive normalization, and a LNL extension of it, neither of which can be represented exactly as an ePPR model.

3.4 Complex cell

In the previous section we showed that ePPR was able to characterize a simulated complex cell from its responses to either natural or random data. We now use the method to characterize a real complex cell recorded from an anesthetized cat. Experimental procedures are described elsewhere (Felsen et al., 2005).

A unique characteristic of the data recorded by Felsen et al. (2005) is that individual cells were probed with both natural and “matched” random stimuli (Section 3.7.1). Below we first characterize the complex cell from its responses to the natural stimuli, and then from its responses to the matched random stimuli.

To determine the memory of the cell, we first estimated ePPR models without time interaction. Then, because the memory was short, and the amount of data large, we estimated models with time interaction. In this section we present models with time interaction, and Figure 3.11 (Appendix 3.8.4) shows a model without time interaction estimated from responses to natural stimuli.

To assess the dependence of ePPR estimates on the size of the training dataset, we estimated models using sets of 3,000, 10,000, and 20,000 responses. For each of these sets, 5 ePPR models were estimated from different fitting subsets (Section 3.7.4). For simplicity, we show the parameters of example models estimated from a fitting subset of the training dataset with 20,000 responses, but the number of terms and correlation coefficients plots show averages across the 5 estimated models.

3.4.1 Data from natural images

We recorded responses to four repeats of the quasi-natural sequence ensemble (Section 3.7.2). The mean total number of spikes in these four sets of responses was 87,258. We used the mean of these sets of responses to estimate the parameters of the different models.

The filters of the example ePPR model estimated from natural data (Figure 3.6a) are consistent with previous estimations of linear subspaces of complex cells from responses to two-dimensional natural (Touryan et al., 2005; Rapela et al., 2006), or random (Movshon et al., 1978a; Chen et al., 2007) images. In particular, the three middle filters have clear Gabor shapes with similar orientation and spatial frequency, but are shifted in phase. However, note that the bottom frame of the rightmost filter (operating on the image presented between 85 and 126ms prior to the cell response) is cross-oriented with respect to the other filters.

For comparison, we estimated a set of 6 MID filters (Section 3.7.9) using the training dataset with 20,000 responses (Figure 3.7). Only the first MID filter (leftmost filter in Figure 3.7a) is well structured, and this filter is similar to the most relevant ePPR filter, according to the β coefficient (leftmost filter in Figure 3.6a). That ePPR recovered more filters with good structure than MID could be explained as a flaw of ePPR in estimating spurious filters, or as a flaw of MID in failing to recover relevant filters. We will return to this point below.

Figure 3.6b shows the nonlinear functions of the example model estimated from natural data. The leftmost nonlinear function is approximately a half-wave rectification. The three middle nonlinear functions, corresponding to the filters with clear Gabor shapes in Figure 3.6a, are full-squared, in agreement with the polarity invariance of complex cells (Movshon et al., 1978a). That complex cells can be characterized with half-wave and full-squared nonlinear functions has been previously reported (Rust et al., 2005, Figure 5). And the rightmost nonlinear function, corresponding to the filter with a cross-oriented frame at long delays, is suppressive, revealing cross-oriented inhibition in the response of this complex cell.

As for the simulated cell, we compared the predictive power of filters estimated by ePPR, MID, nSTC, and PPR, using a second-order multi-dimensional polynomial as the predictive method (Section 3.7.12). Figure 3.6f plots the correlation coefficients between the complex cell responses and the polynomial models predictions, as a function of the

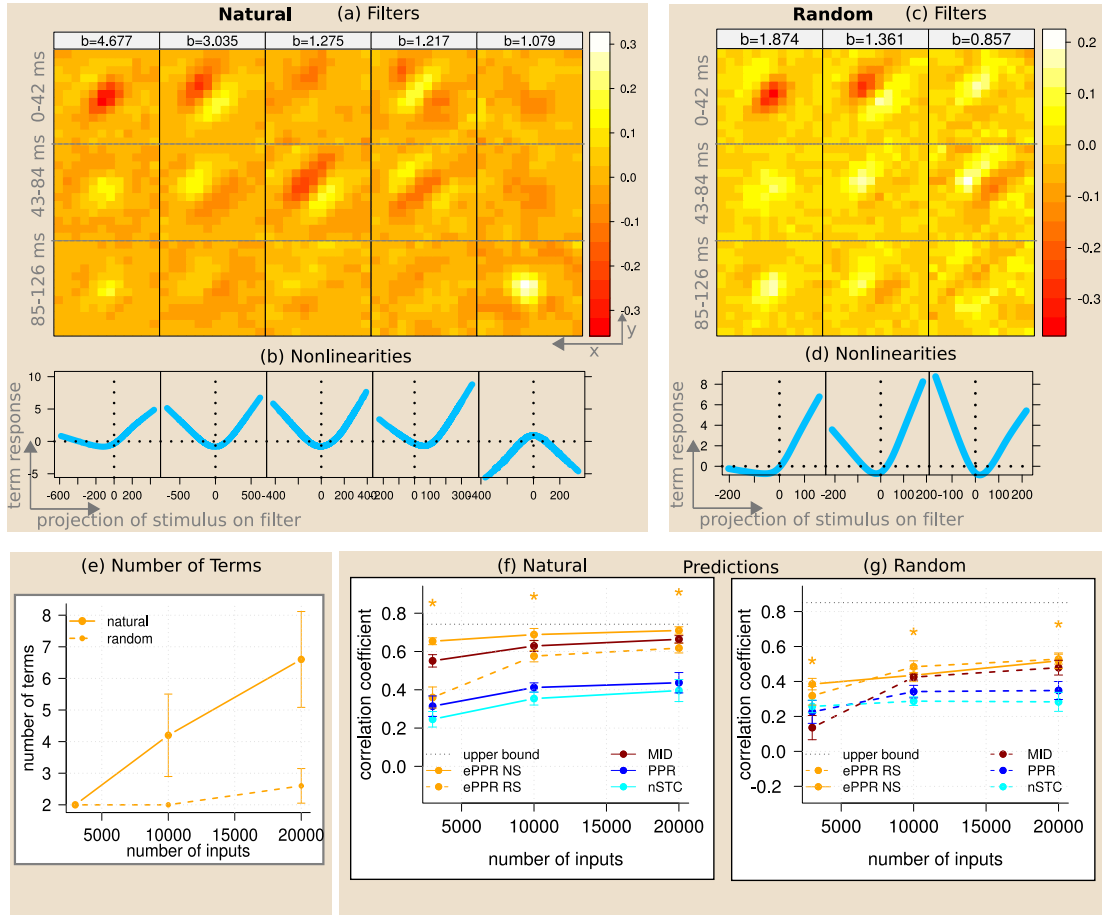


Figure 3.6: Complex cell: ePPR models. (a,b): filters (a) and nonlinear functions (b) of the example model estimated from natural data. The titles in (a) are the corresponding β coefficients. (c,d): as (a,b) but for models estimated from random data. (e): average number of terms in ePPR models estimated from natural and random data. (f-g) predictive power of filters estimated by different methods with same format as in Figure 3.4. The estimated filters and nonlinear functions are consistent with those estimated using previous methods. Models estimated from natural and random data are similar to each other. However, late suppression is only present in the model estimated from natural data. Furthermore, models estimated from natural data recovered more filters than models estimated from random data. For natural data, ePPR filters yielded predictions substantially better than those estimated by other methods, and predictions from ePPR filters were close to the upper bound on the predictive power of any model. For random data, ePPR and MID filters estimated from 20,000 stimuli were very similar (cf. Figures 3.6c and 3.7b) and their predictions were not statistically different.

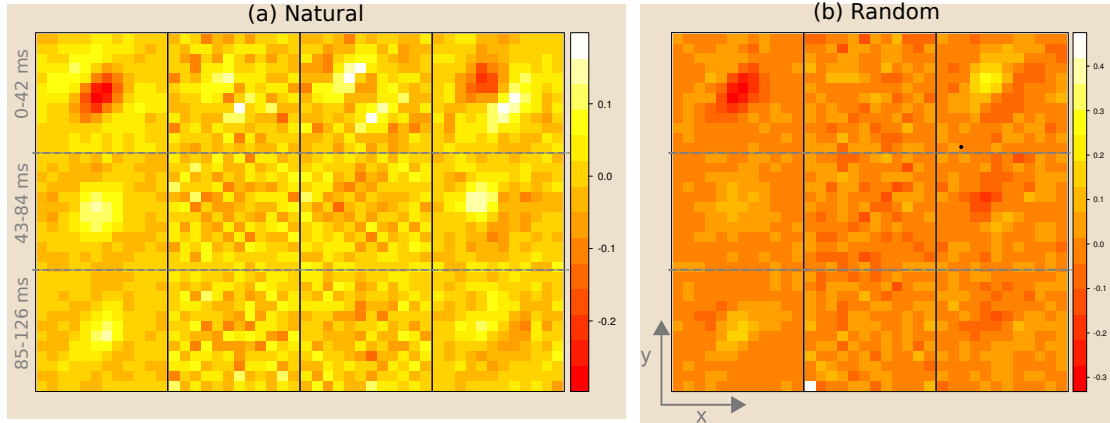


Figure 3.7: Complex cell: example set of MID filters estimated from 20,000 responses. Same format as in Figure 3.3. For natural data (a), only the first filter is well structured. For random data (b), the three filters are well structured, and are very similar to those estimated by ePPR (Figure 3.6c).

number of stimuli used to estimate the filters and polynomial models. The dashed line is an upper bound on these correlation coefficients (Section 3.7.7). For all number of stimuli best predictions are obtained with ePPR filters. Moreover, ePPR filters estimated using 20,000 inputs closely approximate the upper bound on the correlation coefficients. This indicates that the filters with good structure estimated by ePPR and not by MID (cf. Figures 3.6a and 3.7a) are relevant for the predictions of the model, and therefore not spurious.

Figure 3.11c (Appendix 3.8.4) replots the correlation coefficients of the ePPR polynomial model with the correlation coefficients on ePPR models with and without time interactions. It shows that ePPR models with time interaction predict better, or at the same level, as polynomial models, and substantially better than ePPR models without time interaction. Thus, nonlinear interactions between pixels of images at different delays are relevant for the response of this complex cell.

3.4.2 Data from random images

We recorded responses to two repeats of the quasi-random sequence ensemble (Section 3.7.2). The mean total number of spikes in these responses was 73,022.5. We used the mean of these sets of responses to estimate parameters of the different models.

Figure 3.6c shows that the filters obtained with random stimuli are a subset of those obtained with natural stimuli (Figure 3.6a). The leftmost filter estimated from random stimuli corresponds to the leftmost filter estimated from natural stimuli, the second (from the left) filter estimated from random stimuli corresponds to the second filter estimated from natural stimuli, and the third filter estimated from random stimuli corresponds to the fourth filter estimated from natural stimuli (but they are reversed in polarity). Figure 3.6d shows that the nonlinear function obtained with random stimuli correspond to those obtained from natural stimuli (Figure 3.6b).

For comparison, Figure 3.7b shows a set of 3 MID filters (Section 3.7.9) estimated from 20,000 responses. All of these filters have good structure, and are very similar to those estimated by ePPR. This shows that our implementation of the orthogonalization procedure recommended by Dr. Sharpee for the estimation of multiple MID filters (Section 3.7.9) is correct. Also, the similarity between the ePPR and MID filters suggests that the filters estimated by these methods are not spurious.

As expected, because ePPR and MID estimated similar filters from datasets with 20,000 responses, the predictions of the second-order multi-dimensional polynomial models estimated from ePPR filters were not better than those from MID filters (Figure 3.6g, rightmost points in orange and red curves, Wilcoxon signed-rank test, $p > 0.05$, Section 3.7.13). However, for the smaller training datasets, ePPR filters predictions were significantly better than those of MID filters ($p < 0.01$, for 10,000 and 3,000 stimuli). As explained in Section 3.7.7, because we recorded responses to only two repeats of random stimuli, the calculated upper bound is loose.

Two important differences emerged between the models estimated from natural and random data: First, a filter with late suppression was recovered from natural, but not

from random, data. This difference was not an idiosyncrasy of the example models shown, but was present in most ePPR models estimated with more than 3,000 responses (all 5 models estimated from 20,000 natural responses, 3 of the 5 models estimated from 10,000 responses, but no model estimated from random responses, showed late suppression). Second: models estimated from natural data had more filters than those estimated from random data (Figure 3.6g).

In summary, in this section we demonstrated the feasibility of ePPR to characterize a complex cell from its responses to natural and random stimuli. We showed that estimated ePPR models had several features in common with the energy model of complex cells, that ePPR models estimated from natural and random stimuli were very similar to each other, but displayed a few interesting differences, and that predictions from ePPR models were close to an upper bound on the predictive power of any model.

3.5 Simple cell

The ePPR model is general and one can use it to characterize a wide variety of visual cells. Here, we test this generality by using ePPR to model a cortical simple cell (Felsen et al., 2005) from its responses to both natural and random stimuli. We follow the same procedure as in Section 3.4.

For natural stimuli, we recorded responses to two repeats, obtaining two sets of responses with a mean total number of spikes of 8,753. For random stimuli we also recorded responses to two repeats, obtaining two sets of responses with a mean total number of spikes of 6,673.5.

Figure 3.8 shows that the estimated filters and nonlinear functions follow the same pattern as for the complex cell (Figure 3.6). Natural and random stimuli yield similar

filters and nonlinear functions. Consistently with earlier data, the filters are Gabor-like and oriented, and their nonlinear functions are half-wave rectifications (Movshon et al., 1978b). However, two important differences are evident: First, as for the complex cells, explaining responses to natural stimuli requires more filters (compare Figures 3.8a and 3.8c, see also Figure 3.8e). Second, the model estimated from random data, with an estimated filter that has frames with similar shape at all delays, but whose amplitudes are modulated in time (Figure 3.8c), and with a half-wave rectification, (Figure 3.8d), resembles the classical space-time separable model for simple cells (DeAngelis et al., 1993b; DeAngelis, Ohzawa, & Freeman, 1993a). In contrast, the model estimated from natural data has mixed features of simple and complex cells. For this model, the most important term according to the β coefficient (left column in Figures 3.8a and 3.8b) is also typical of a simple cell. However, the least important term (right column in Figures 3.8a and 3.8b) is consistent with the energy model for complex cells (Adelson & Bergen, 1985): The second filter has a similar shape as the first filter, but it is shifted in phase, and its nonlinear function is fully rectified. Therefore, if we look only at the most important term of the model estimated from natural data, we see a typical space-time separable simple cell. But, if we look at all the terms, we see a mixture of a simple and a complex cell.

For filters estimated from 20,000 natural stimuli, third-order polynomials (Section 3.7.12) generated significantly better predictions with ePPR filters than with MID filters (Wilcoxon signed-rank test $p < 0.01$, Figure 3.8f, Section 3.7.13). Also, for 20,000 and 10,000 natural stimuli, predictions from ePPR filters were significantly better than those from regularized spike triggered average (rSTA; Smyth et al., 2003, Section 3.7.11) ($p < 0.01$, Figure 3.8f). And for all number of natural stimuli ePPR filters yielded better predictions than PPR filters⁴ ($p < 0.01$). For random stimuli all four models predicted similarly (Figure 3.8g).

⁴PPR models were estimated using the images presented to the cell at the time bin before the response of the cell.

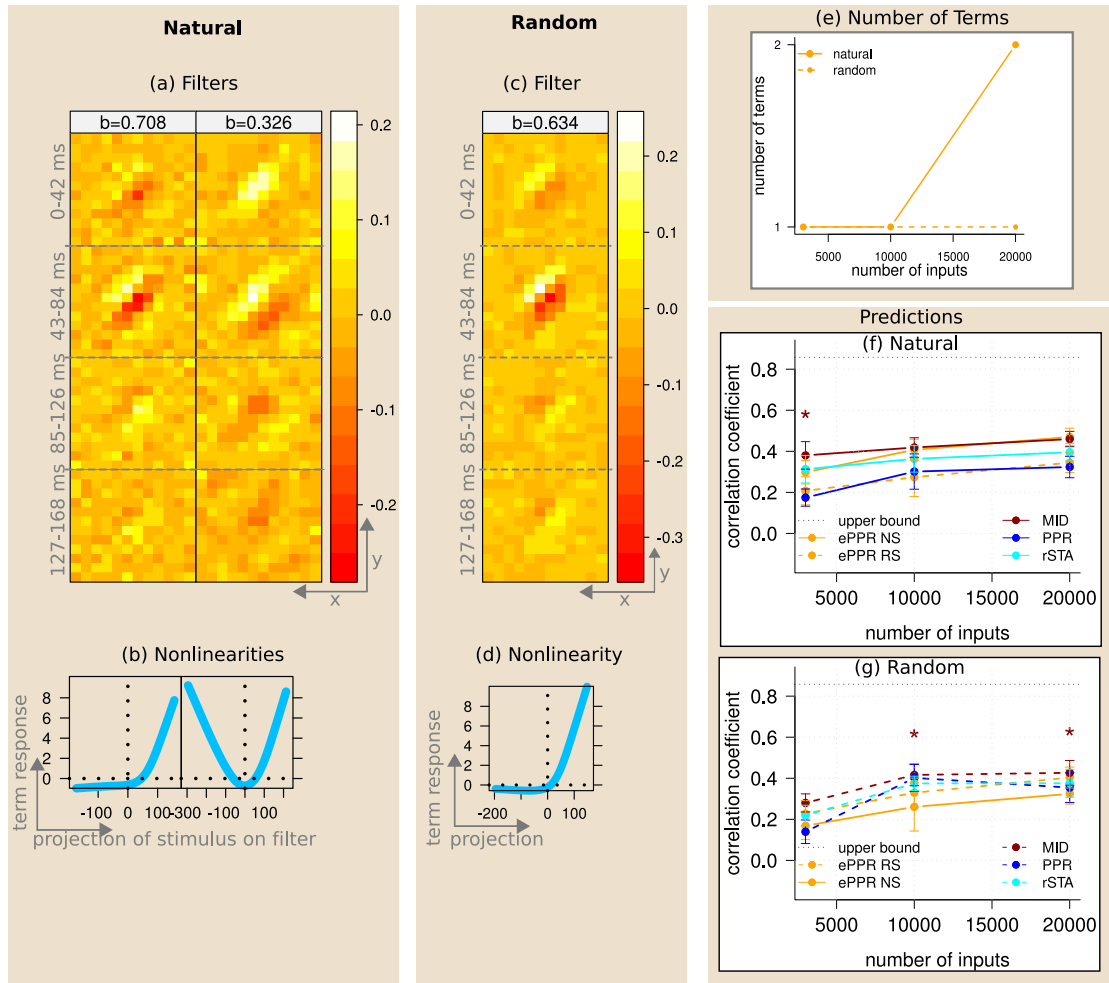


Figure 3.8: Simple cell: ePPR models. The format of this figure is identical to that of Figure 3.6, but the cyan curves in (f,g) correspond to polynomial models constructed with rSTA filters. The figure in (e) does not contain error bars because, for each number of inputs, all estimated models had the same number of terms. The estimated filters and nonlinear functions are consistent with those estimated using previous methods. Models estimated with natural and random data are similar to each other. However, the model estimated from natural data, but not that estimated from random data, has features of a complex cell model. Furthermore, models estimated from natural data recovered more filters than those estimated from random data. For natural stimuli, predictions from ePPR filters were better or equal than those of previous methods. For random stimuli the filters estimated by the four methods gave similar predictions.

In summary, in this section we demonstrated the feasibility ePPR to characterize a simple cell from its responses to natural and random image sequences. We showed that estimated ePPR models had several features in common with the standard space-time separable model for simple cells. Also, ePPR models estimated from natural and random data were very consistent with each other but, as for the complex cell, had a few interesting differences.

3.6 Discussion

In this study, we demonstrated the feasibility of ePPR for the spatio-temporal characterization of visual cells. ePPR is the first algorithm to apply a projection pursuit estimation strategy for the spatio-temporal characterization of visual cells from input/output data. This strategy addresses a fundamental problem in modeling visual cells from input/output data: the curse of dimensionality (Bellman, 1961). By overcoming this problem, ePPR was able to estimate, for the first time, a very general linear-nonlinear model that is spatio-temporal, uses two-dimensional images as inputs, and contains multiple filters.

The ePPR model can approximate a broad class of cells. In Section 3.2 we proved that ePPR models can uniformly approximate any continuous function to an arbitrary degree of precision. We validated this generality empirically. First, due to the divisive normalization, the model of the simulated complex cell (Equation 3.8) cannot be exactly represented by an ePPR model (Equation 3.6). Nevertheless, ePPR approximations were very good (Figure 3.2), and these approximations remained good for different strengths of divisive normalizations (Appendix 3.8.5). Second, interactions between pixels of images at different delays are relevant to the responses of the simulated complex cell (Equation 3.8). These interactions are neglected by ePPR models without time interaction. In spite of this, ePPR models without time interaction provided good approximations (Figure 3.10, Appendix 3.8.4). Also, interactions between pixels of images at different delays

were relevant for the cortical complex cell, as evidenced by the substantially better predictions of the ePPR model with time interaction over those of the ePPR model without time interaction (orange versus pink curves in Figure 3.11c). For this cell ePPR models without time interaction were very similar to the ePPR models with time interactions (whose predictive power was very close to an upper bound on the predictive power of any model). And third, ePPR provided a reasonable approximation to a linear-nonlinear-linear extension of the simulated complex cell model, Figure 3.5.

Since ePPR makes no assumptions about the statistical properties of its inputs, it is well-suited to characterize visual cells from their responses to arbitrary, including natural, inputs. We tested this with simulated and physiological data. For the simulated cell we showed that ePPR models estimated from natural and random data well matched the simulated cell model (Figure 3.2). And for the cortical complex and simple cells, we showed that models estimated from natural data were very consistent with those estimated from random data, and with previous characterizations of these cells (Figures 3.6 and 3.8).

Because ePPR is an entirely non-parametric algorithm, and because the cell responses were very noisy, ePPR estimates could have been very variable. However, this was not the case (Figures 3.2d, 3.2g, 3.6e, and 3.8e). Furthermore, ePPR models estimated from natural and random data, i.e. with stimuli and responses with very different statistics, were very similar to each other for the simulated cell (Figure 3.2), for the complex cell (Figure 3.6) and for the simple cell (Figure 3.8). Several features of the ePPR estimation procedure help reduce the variability of the estimated parameters. First, the projection pursuit strategy used by ePPR reduces the original large-dimensional problem of estimating all the parameters in the ePPR model, Equation 3.6, to a sequence of low-dimensional problems of alternatively minimizing the criterion in Equation 3.7, first with respect to the nonlinear function ϕ , and then with respect to the filter, α . Second, the optimization criterion for the filters, α in Equation 3.7, penalizes non-smooth filters. And third, the

estimation of the nonlinear functions, ϕ in Equation 3.7, is performed using smoothing splines with a relatively large penalty for non-smooth estimates.

Parameters of estimated ePPR models displayed properties previously reported in the literature. For both the simple and complex cells, estimated filters had an oriented Gabor-like shapes with similar spatial frequencies and orientations (Movshon et al., 1978b, 1978a). For the simple cell, different frames of estimated filters had a very similar shape, but their amplitude was modulated in time, in agreement with the space-time separable model for simple cells (DeAngelis et al., 1993b, 1993a). For the complex cell, inhibition appeared late in time, as previously reported (Rust et al., 2005; David et al., 2004). For the complex cell, most nonlinear functions were full-squared rectifications (Figure 3.6b, and 3.6d), while for the simple cell they were half-wave rectifications (Figure 3.8b, and 3.8d).

Although the main goal of this article was to demonstrate the feasibility of ePPR, the results presented here suggest that the response properties of cortical cells may depend on the statistics of the stimuli used to probe them. Recently, Sharpee et al. (2008) showed that spatio-temporal LN models of simple cells cells, estimated from natural and random stimuli, displayed significant differences. However, these LN models contained only one filter. Here, we showed, for the first time, differences in spatio-temporal LN models, with multiple filters, estimated from natural and random stimuli. We found that ePPR recovered more filters from natural than from random data, for the complex cell (Figure 3.6f), for the simple cell (Figure 3.8f), but not for the simulated cell (Figure 3.2h). Also, ePPR recovered inhibitory terms at later delays in the models of the complex cell estimated from natural, but not from random, data, consistent with the findings of David et al. (2004). Furthermore, the simple cell models estimated from natural responses, but not those estimated from random responses, displayed properties typical of complex cells, supporting the notion by Mechler and Ringach (2002) that simple and complex cells are not two different classes of cells but lie in a continuum. This study thus provides further support for the notion that the observable response properties of sensory cells

depend on the statistics of the stimuli used to probe them (Woolley, Fremouw, Hsu, & Theunissen, 2005; Wooley et al., 2006; Theunissen et al., 2000; David et al., 2004; Felsen et al., 2005; Sharpee et al., 2006, 2008; David, Mesgarani, Fritz, & Shamma, 2009). However, differences in parameters of models estimated from responses of cells to stimuli with different statistics should be interpreted with care. These differences could be artifacts of using overly-constrained models (Christianson, Sahani, & Linden, 2008). But this does not appear to be the case for ePPR and the cells characterized in this article. Because the ePPR model with time interactions can uniformly approximate, to an arbitrary degree of precision, any continuous function with inputs in the unit cube (Section 3.2), it is not overly-constrained to represent a large class of cells. For instance, due to the divisive inhibition, the simulated complex cell characterized in Section 3.3 could not be represented exactly by an ePPR model. But, despite this constraint, ePPR models estimated from natural and random stimuli were very similar to each other (Figure 3.2).

We compared ePPR with state of the art methods for the estimation of linear-nonlinear models, and ePPR performed better, or at the same level, as these methods. From the cells we analyzed in the dataset from Felsen et al. (2005), the cortical complex cell shown in this article is the ones for which we obtained the best correlation coefficient. However, similar correlation coefficients, and qualitatively similar ePPR estimates, were obtained for other complex cells (Figure 3.14, Appendix 3.8.7).

The method with results more similar to ePPR was MID. The estimation of multiple filters is the most important advantage of ePPR over MID. For the simulated cell ePPR well approximated the three-dimensional structure of the true filter space (Figure 3.2b, orange curve in Figure 3.2d, Figure 3.2e, and orange curve in Figure 3.2g), while MID well approximated the true filter space only along one dimension (Figure 3.3a, red curve in Figure 3.2d, Figure 3.3b, and red curve in Figure 3.2g). And from responses of the real complex cell to natural stimuli ePPR recovered six filters (Figure 3.6a), while MID recovered only one well-structured filter (Figure 3.7a), and the predictive power of ePPR filters was substantially better than that of MID filters (Figure 3.6f). But, this advantage

of ePPR over MID does not always hold. From responses of the real complex cell to random stimuli ePPR and MID recovered very similar sets of three filters (Figure 3.6c and Figure 3.7b). Note that the sequential procedure that we used to estimate multiple MID filters (Section 3.7.9) is not optimal for natural stimuli. The optimal procedure requires the joint estimation of all the filters. This is a very challenging problem. For example, to jointly estimate the five filters of the complex cell would require the estimation of a five-dimensional probability distribution. If this estimation is done using histograms, as in the current implementation of MID, it will be almost impossible to record enough physiological data to accurately sample all the bins in a five-dimensional histogram, and obtain reliable estimates of the joint probability distribution. Currently an implementation of a procedure to jointly estimate MID filters is not publicly available. Eventually, if this implementation becomes available, it will be important to compare the new MID filters with those from ePPR.

The greedy estimation of the forward model by PPR is not guaranteed to converge to the global optimum. For this reason PPR uses the backward procedure to discard spurious terms from the forward model. However, for responses with large amounts of noise, or models estimated with small data sets, the backward procedure does not remove all spurious terms from the forward model. So, in ePPR we use a model selection procedure to select the best model from the collection of models returned by the backward procedure. Still, the backward and model selection procedures are heuristic and do not guarantee convergence to the global optimum. To study how frequently ePPR returned suboptimal solutions, for the simulated and cortical cells studied in this article we estimated five ePPR models from different subsets of the training dataset (Section 3.7.4). For data sets with intermediate or low noise, models of the simulated cell were all very good approximations, with the correct number of filters (Figure 3.2h), which were very similar to the true filters and had small variability (Figures 3.2b, 3.2d, 3.2e, and 3.2g), and with nonlinear functions with correct shape and sign (Figures 3.2c and 3.2f). For large data sets, models of the complex cell estimated from natural data approached an upper bound on the predictive

power of any model (Figure 3.6a). So, for models estimated from good quality data, the convergence to suboptimal solutions does not seem to be a severe problem. ePPR models estimated from datasets with large amounts of noise, or with small or intermediate sizes, recovered less terms than the optimal (Figures 3.2h, 3.6e, and 3.8e) but, thanks thanks to a stringent model selection procedure (Section 3.7.6), did not contain spurious terms.

To estimate spatio-temporal models one can use ePPR models with or without time interactions (Section 3.2.2). Models with time interaction are more general than models without time interaction, but require estimating larger spatio-temporal filters (and therefore more training data), and knowing the memory of the cell in advance. In contrast, models without time interaction contain smaller spatial filters, and the estimation algorithm discovers the memory of the cell in only one run. With the exception of the LNL cell (Section 3.3.3), for all the cells studied in the main body of this article –that have been probed with a large stimuli set, and whose response depends on only three or four previous frames– we estimated models with time interaction. But for the simulated LNL cell –whose response depends on movie frames presented up to 9 delays prior to the cell response– and for the simulated cell probed with the temporally correlated natural movie ensemble –for which the temporal correlation reduces the effective number of inputs– we estimated models without time interaction.

The model used in ePPR is that of an artificial neural network with three layers (input layer, hidden layer, and output layer). Thus, ePPR is related to the neural network method proposed by Prenger et al. (2004). Both methods use non-parametric models that can characterize a large variety of visual cells. Also, because both are regression-based methods, they can estimate their parameters using natural stimuli. However, because the neural network method estimates all its parameters simultaneously, it does not overcome the curse of dimensionality. A detailed comparison between projection pursuit and neural network methods is given in Hwang et al. (1994). To bypass the curse of dimensionality Prenger et al. (2004) used projections of the input images in a few principal components as inputs to the neural network. But, important information about the inputs could be

lost in these low-dimensional projections. The strategy used by ePPR to bypass this problem is a more general one.

Below we summarize some advantages and disadvantages of the ePPR algorithm.

Advantages

1. *Generality.* As discussed above, the ePPR model is very general and can approximate many classes of cells.
2. *Finding the model structure in only one estimation.* Provided the number of terms per delay, M_d^L , and the number of delays, D^L , of the forward model are sufficiently large, only one estimation of an ePPR model without time interactions finds the model structure, i.e., number of delays and number of terms for each delay, required to characterize a given cell. This feature is nicely illustrated by the ePPR model estimated for the LNL simulated cell (Section 3.3.3).
3. *Natural stimuli.* As discussed above, ePPR models can be estimated using natural stimuli. This is an advantage over previous methods that require Gaussian stimuli (Chichilnisky, 2001; Rust et al., 2005).
4. *Temporal dimension.* Previous methods (Touryan et al., 2005; Rapela et al., 2006; Chen et al., 2007) used only one image in the past to predict the current response, while ePPR uses several past images. As shown in Figures 3.4, 3.6f, and 3.6g, spatio-temporal ePPR models predicted responses substantially better than purely spatial PPR and nSTC models.
5. *Spatially 2D model.* Rust et al. (2005) estimated a spatio-temporal LN model visual cells. However, to reduce the number of parameters in their estimations, the input images varied along a single spatial dimension (optimally-oriented bars). Due to its efficient optimization algorithm, and to the smooth prior for its filters (Section 3.2.3), ePPR allows the use of two-dimensional images.

6. *Multiple filters.* We have shown that for the simulated cell (Figure 3.2), the complex cell (Figure 3.6), and the simple cell (Figure 3.8), ePPR recovered more than one filter. This contrasts with other methods that can only estimate one filter (Chichilnisky, 2001).
7. *Suppressive filters.* ePPR successfully recovered the suppressive filter of the simulated cell (Figure 3.2), and a suppressive filter from responses of the cortical complex cell to natural stimuli (Figure 3.6). To our knowledge, ePPR is the first non-parametric algorithm that has been shown to estimate suppressive filters from responses of visual cells to natural stimuli.
8. *Improved predictions.* ePPR provided better predictions of the responses of the simulated cell (Figure 3.4), the complex cell (Figure 3.6f, and 3.6g), and the simple cell (Figure 3.8f) than previous methods.

Disadvantages

1. *Spurious terms.* The analysis of projection pursuit techniques by Huber (1985) concludes with a nice description of this limitation:

Perhaps the practical conclusion to be drawn is that we shall have to acquiesce to the fact that Projection Pursuit will in practice uncover not only true but also spurious structure, and that we must weed out the latter by other methods, for example by validating the results on different data sets.

To weed out the spurious structure from ePPR models we used a model selection procedure based on cross-validation (Section 3.7.6). With data sets large enough to perform reliable cross-validations, our model selection procedure worked very well. However, large recordings are not very common in visual neurophysiology. So we are currently investigating Bayesian model selection procedures to perform model selection without the need of cross-validation data.

2. *Hyper-parameters.* As shown in Section 3.7.8, ePPR contains many hyper-parameters. We used reasonable heuristics to select their values (Section 3.7.6), but we would like to learn them from data, as done in Roosen and Hastie (1994) with the degrees of freedom of the splines.
3. *Nonlinearities.* The ePPR estimation algorithm avoids the curse of dimensionality by working with one-dimensional projections. The cost is that it is poorly suited to characterize highly nonlinear functions (Huber, 1985).
4. *Global convergence.* As discussed above, ePPR is not guaranteed to converge to the global optimum. However, this is a problem shared by all non-parametric nonlinear optimization techniques.
5. *Speed.* The ePPR estimation algorithm is iterative, requiring several fits and refits of models with different numbers of terms, and is thus very slow. On a personal computer, with a 3GHz processor and 2GB of memory, estimating the example ePPR model for the cortical complex cell from natural data requires approximately 45 minutes. The low speed of ePPR contrast with the high speed of nSTC. Spatial nSTC models can be estimated in seconds. However, ePPR is fast compared to MID. Estimating the five MID filters of the complex cell from natural data, using the sequential approach, requires approximately 25 hours.

In conclusion, this article has demonstrated the feasibility of ePPR, a very general method for the spatio-temporal characterization of visual cells from arbitrary (including natural) stimuli, and showed that ePPR compared favorably with information-theoretic and spike-triggered techniques.

3.7 Methods

3.7.1 Overview of the experimental paradigm

The cortical complex and simple cells analyzed here were subjected to the following sequence of inter-dependent experiments, as described in Felsen et al. (2005).

1. Responses from a cell were recorded to a quasi-natural sequence ensemble (see Section 3.7.2).
2. The filter(s) of the cell were estimated using the responses to natural images recorded in Step 1.
3. An ensemble of random images, matched to the ensemble of natural images used in Step 1, was constructed (Section 3.7.2).
4. Finally, responses from the cell in Step 1 were recorded to several interleaved repeats of the random and a novel natural ensemble.

3.7.2 Stimulus ensembles

The following stimulus ensembles were used to characterize the simulated and cortical cells.

Natural movie : This ensemble consists of the center patches (16×16 pixels) extracted from a digitized natural movie (van Hateren & Ruderman, 1998). It has the spatial and temporal correlation structure present in natural movies. This ensemble was used to probe the simulated cell in Appendix 3.8.6.

Natural sequence : This ensemble is identical to the natural movies ensemble, with the exception that the frames in the movie have been randomly resorted. Thus, it contains the spatial, but not the temporal, correlation in natural movies. This ensemble was used as the natural stimuli to probe all simulated cells, with the exception of that in Appendix 3.8.6.

Random sequence : Each frame of this ensemble was matched to the corresponding frame in the natural sequence ensemble, such that a) the matched frames had the same mean and root-mean-square contrast, and b) the dot products of each true filter of the simulated the cell onto the matched frames were equal. Random images contain no spatial structure and are visually indistinguishable from white noise. However, due to the matching procedure, they are not white noise. Further details of the matching procedure are provided in Felsen et al.(2005). This ensemble was the random stimuli used to probe all simulated cells.

Quasi-natural sequence : Raw images were selected at random from a database consisting of a variety of digitized natural movies (van Hateren & Ruderman, 1998), and the center patch (12×12 pixels) of each image was retained. These patches were normalized to have the same root-mean-square contrast and, to maximize their diversity, one patch of each a pair of very similar patches was excluded from the ensemble. Further details are given in Felsen et al.(2005). This was the natural ensemble used to probe the cortical complex (Section 3.4) and simple (Section 3.5) cells.

Quasi-random sequence : This ensemble was constructed in the same way as in the random sequence ensemble, but with frames matched to the quasi-natural sequence ensemble. It was used to probe with random stimuli the cortical complex (Section 3.4) and simple (Section 3.5) cells.

3.7.3 Simulated responses

To study the dependence of ePPR estimates on the noise level of the responses (Figures 3.2, 3.4), we generated three sets of responses with different noise levels. In Equation 3.8, the constant γ controls the mean, and noise, of the responses. We set γ so that at the lowest, intermediate, and highest noise levels the mean of \bar{y} in Equation 3.8 was $E[\bar{y}] = 5.62, 0.56, \text{ and } 0.17$ spikes/image, respectively. The intermediate noise level

was selected so that the mean correlation coefficient between simulated responses and ePPR model (without time interactions) predictions was similar to that of the cortical complex analyzed in Section 3.4 (0.58 for the simulated cell versus 0.61 for the complex cell). Simulated responses to random stimuli were generated using the same parameters as for responses to natural stimuli.

We simulated a LNL model by filtering the frame rates generated by the above model with intermediate amount noise with a linear-phase lowpass filter with a cutoff frequency of $\pi/2$ cycles per sample and with a length of 15 samples. The output of this filter at time n was the mean of a Poisson random variable giving the number of spikes at time n . These simulated responses were used to estimate the models in Figure 3.5.

To assess the dependence of ePPR estimates on the amount of divisive inhibition (Figure 3.12), we fixed the value of the γ parameter, to the value used above to generated responses with intermediate amount of noise, and varied the inhibitory constant ω so that the mean of the denominator in Equation 3.8 was 4.26, 20.58, and 40.17.

To examine the effect of temporal correlations on ePPR estimates (Figure 3.13) we simulated responses to the natural movie ensemble (Section 3.7.2) with the parameters used above to generated responses with intermediate amount of noise.

3.7.4 Data partitioning

For the simulated and cortical cells, the complete dataset contained 24,000 responses. We generated training datasets of different sizes: 20,000, 10,000, and 3,000 responses. For each training dataset we use a disjoint set of 4,000 responses as testing dataset. Each testing dataset was further partitioned into 8 disjoint testing subsets, with 500 responses each. From each training dataset we generated 5 fitting subsets, by excluding contiguous and disjoint subsets of 20% of the responses of the training dataset. For each fitting subset, the excluded responses made the validation subset. The validation subset was further partitioned into 8 disjoint subsets.

For example, the training subset with 20,000 responses contained responses to images 1 to 20,000. The corresponding testing dataset contained responses to images 20,001 to 24,000. From the training subset we generated 5 fitting subsets. For instance, the first fitting subset contained responses to images 1 to 16,000, and the corresponding validation subset contained responses to images 16,001 to 20,000. The second fitting subset contained responses to images 1 to 12,000 and 16,001 to 20,000, and the corresponding validation subset contained responses to images 12,001 to 16,000.

For a given training dataset, the 5 fitting subsets were used to estimate 5 different ePPR models. For each estimation of an ePPR model, the validation subsets were used to select the final ePPR model, out of those returned by the backward-stepwise procedure, as shown in Section 3.7.6. Finally, the testing subsets were used to assess the predictive power of the final ePPR model.

3.7.5 Similarity between two sets of filters

What matters to determine the similarity between two sets of filters is not the similarity between any pair of filters in the sets, but the similarity between the spaces spanned by the two sets of filters. Call \mathcal{S}_1 and \mathcal{S}_2 the spaces spanned by the sets of filters. If the dimension of \mathcal{S}_1 equals that of \mathcal{S}_2 , then principal angles (Golub & van Loan, 1996) are good measures to study their similarity. For the simulated complex cell, the dimension of the true filter space is three, which equaled the dimensions of the filter spaces estimated by ePPR and MID. So, we used principal angles to study the similarity between the filters of the simulated cell and those estimated by ePPR or MID (Figures 3.2d, 3.2g, and 3.12a). When comparing two subspaces of dimension n one can compute n principal angles. This turned to be an advantage of principal angles respect to single-valued distance measures because, as shown below, the three principal angles of the simulated cell gave us further insight about the quality of the ePPR and MID estimates than what we could have obtained from a single-value distance measure.

Principal angles

Let \mathcal{S}_1 and \mathcal{S}_2 be subspaces of \mathbb{R}^m whose dimensions satisfy

$$p = \dim(\mathcal{S}_1) \geq \dim(\mathcal{S}_2) = q \geq 1$$

The principal angles $\theta_1, \dots, \theta_q \in [0, \pi/2]$ between \mathcal{S}_1 and \mathcal{S}_2 are defined recursively by

$$\cos(\theta_k) = \max_{u \in \mathcal{S}_1} \max_{v \in \mathcal{S}_2} u^T v = u_k^T v_k$$

subject to:

$$\begin{aligned} \|u\| &= \|v\| = 1 \\ u^T u_i &= 0 & i = 1 : k-1 \\ v^T v_i &= 0 & i = 1 : k-1 \end{aligned}$$

Note that the principal angles satisfy $0 \leq \theta_1 \leq \dots \leq \theta_q \leq \pi/2$. The vectors $\{u_1, \dots, u_q\}$ and $\{v_1, \dots, v_q\}$ are called the principal vectors. If the columns of Q_1 and Q_2 define orthonormal bases of \mathcal{S}_1 and \mathcal{S}_2 , respectively, then $\cos(\theta_i) = \sigma_i$, with σ_i the i^{th} singular value of $Q_1^T Q_2$.

The sine of the first principal angle, $\sin \theta_1$, measures the similarity between the closest vectors in \mathcal{S}_1 and \mathcal{S}_2 . Thus, $\sin \theta_1$ can be interpreted as an indication of how well \mathcal{S}_1 approximates \mathcal{S}_2 along one dimension. And for $k > 1$, the sine of the k^{th} principal angle, $\sin \theta_k$, measures the similarity between the closest vectors in the subspace of \mathcal{S}_1 orthogonal to the space spanned by $\{u_1, \dots, u_{k-1}\}$, and the subspace of \mathcal{S}_2 orthogonal to the space spanned by $\{v_1, \dots, v_{k-1}\}$. Then, $\sin \theta_k$ indicates how well \mathcal{S}_1 approximates \mathcal{S}_2 along k dimensions. For the simulated cell, this interpretation of principal angles showed

that MID approximations were good along one dimension but poor along more than one dimension (Figures 3.2d and 3.2g).

3.7.6 Selecting the best ePPR model by cross-validation

The backward-stepwise procedure, Listing 8, returns a set of models. We used a cross-validation procedure to select the best model from this set, as described here. To give a concrete example, we provide details of a cross-validation procedure used to obtain an ePPR model with time interaction estimated from simulated responses with large levels of noise to natural stimuli. For this estimation, the backward-stepwise procedure returned a set of models having between 1 and 6 terms. We then used each of these 6 models to predict responses to the 8 validation subsets (Section 3.7.4). Figure 3.9a plots the correlation coefficients between these predictions and the responses of the simulated cell as a function of the number of terms of the ePPR models. For each number of terms, n , the value j along the y-axis, $1 \leq j \leq 8$, is the correlation coefficient between the responses from the simulated cell, to images in the j th validation subset, and the predictions of the ePPR model with n terms.

We seek to select the model that has the best predictive power, while containing the fewest terms. Thus, the predictions of this model should be better than those of all models with fewer terms, and not worse than those of models with more terms. To compare the predictive power between two models we test, using a non-parametric Wilcoxon signed-rank test, if the correlation coefficients of one of the models are larger/smaller than those of the other model. For the example cross-validation procedure, with the correlation coefficients shown in Figure 3.9a, we found the ePPR model with 4 terms yielded better predictions than all ePPR models with fewer terms ($p < 0.05$ for all Wilcoxon signed-rank tests). Also, we could not conclude that the ePPR model with 4 terms yielded worse predictions than any ePPR model with more terms ($p > 0.1$ for all Wilcoxon signed-rank tests). Thus, to characterize the simulated cell from natural data, the best ePPR model contains 4 terms. The filters of this model are shown in Figure 3.9b.

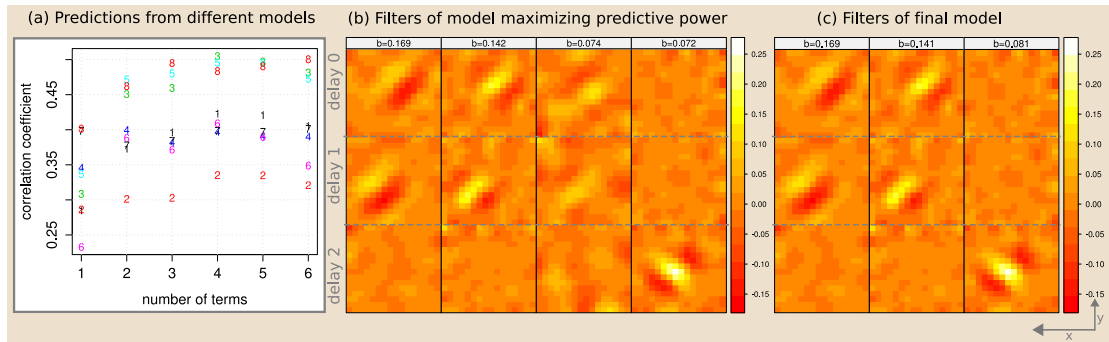


Figure 3.9: Selection of the best ePPR model by cross-validation. (a) correlation coefficients between ePPR model predictions and cell responses as a function of the number of terms in ePPR models. For each number of terms, n , the value j along the y-axis, $1 \leq j \leq 8$, is the correlation coefficient between the responses from the simulated cell to images in the j th validation subset, and the predictions of the ePPR model with n terms. The ePPR model with 4 terms predicts better than all models with smaller number of terms ($p < 0.05$ for all Wilcoxon signed-rank tests). We could not conclude that the ePPR models with 4 terms predicted worse than any model with more than 4 terms ($p > 0.1$ for all Wilcoxon signed-rank tests). Thus, the best ePPR model contains 4 terms. (b): filters of the ePPR model with 4 terms. The third filter from the left is spurious. For the simulated cell, spurious filters appeared only in models estimated from responses with the largest noise level. They were removed from the model using the “Removal of spurious terms” procedure. (c): final ePPR estimate obtained by applying the “Removal of spurious terms” procedure to the ePPR model with 4 terms.

For responses with high levels of noise, the ePPR model maximizing the predictive power can contain spurious terms (e.g., third filter from the left in Figure 3.9b). To remove these spurious terms we use the following procedure.

Removal of spurious terms

In the plot of correlation coefficients versus number of ePPR terms (Figure 3.9a), we see that before reaching the number of terms at which the predictive power of ePPR models saturates (4 terms in Figure 3.9a), in most cases, when the backward-stepwise procedure drops a term from a model with n terms, it yields a new model with $n - 1$ terms having worse predictive power. This means that the term dropped by the backward-stepwise procedure contributed to improve the predictions of the model with n terms. However, occasionally, the model with $n - 1$ terms has equal or better predictive power than the model with n terms (e.g., $n=3$ in Figure 3.9a). This means that the term dropped by the backward-stepwise procedure did not contribute to improve the predictions of the model with n terms. With simulated data we verified that these dropped terms are spurious ones. Then, to detect spurious filters, for each number of terms, n , between 2 and the maximum number of terms, we test whether the correlation coefficients of the ePPR model with n terms are significantly larger than those of the ePPR model with $n - 1$ terms (Wilcoxon signed-rank test, $p < 0.05$). If the test does not reach significance, the term dropped from the model with n terms is removed from the model maximizing the predictive power.

In the example cross-validation procedure considered above, the correlation coefficients of the models with 3 terms were not significantly larger than those of the models with 2 terms (Wilcoxon signed-rank test, $p > 0.1$). We therefore removed from the ePPR model with 4 terms (Figure 3.9b) the term dropped by the backward-stepwise procedure from the models with 3 terms, obtaining a final model which filters are shown in Figure 3.9c.

Note that the procedures that we use to select the ePPR model maximizing predictive power and to remove spurious terms depend only on the outcomes of hypothesis tests, and not on user-dependent subjective criteria. With simulated data, these procedures enabled us to recover the true model, even under considerably noisy conditions. And with physiological data, for models estimated from different fitting subsets, or from natural and random data, this procedure produced very consistent results.

3.7.7 Upper bound on correlation coefficients for predictions

For the simulated responses, for which we knew the firing rate, $r = \bar{y}$ in Equation 3.8, we used as the upper bound the correlation coefficient between the noisy responses, y , and the firing rate, `upperbound` = $\rho(r, y)$.

For the responses of the real cell, for which we did not know the firing rate, we estimated it as the mean response of the M repeats of N stimuli, $\hat{r}(i) = \frac{1}{M} \sum_{m=1}^M y_m(i)$, $1 \leq i \leq N$, and then used as the upper bound the maximum correlation coefficient between the estimated firing rate and the response to any repeat of the stimuli, `upperbound` = $\max_{1 \leq m \leq M} \rho(\hat{r}, y_m)$. Note that when the number M of repeats is small, as in the complex cell characterized with random data, or the simple cell, the mean of the M repeats will be a poor estimate of the firing rate, and the resulting upper bound will be loose (Figures 3.6g, 3.8f, 3.8g).

3.7.8 ePPR hyper-parameters

The only ePPR hyper-parameter (Section 3.8.2.2) that we selected (by cross-validation) for each ePPR estimation is the regularization parameter for the filters, λ . The memory of the forward model, D^L , and the number of terms at delay d of the forward model, M_d^L , were selected by cross-validation for each cell, and kept constant across different conditions, i.e., noise levels, number of stimuli, or type of stimuli. The values selected for these parameters for the example cells shown in this article appear in Table 3.1. The remainder

Figure	λ	M_d^L
3.2b	15	6
3.2e	15	6
3.6a	75	10
3.6c	75	10
3.8a	10	10
3.8c	75	10
3.10a	15	6,6,4,2
3.11a	75	6,6,6,3
3.12b	15	6
3.13a	15	6,6,4,2
3.13b	150	6
3.14b	3	3,6,6,4,3,3

Table 3.1: ePPR hyper-parameters for the cells analyzed in this article. Only shown are the values of the regularization parameter, λ , and number of terms per delay for the forward model, M_d^L . The remainder hyperparameters were fixed for all the ePPR estimations (see text). $M_d^L = (i, j, k, \dots)$ stands for i terms at delay 0, j terms at delay 1, k terms at delay 2, \dots

ePPR hyper-parameters were kept constant for all the cells analyzed in this article ($d=5$, `addTermsCV=0.01`, `refitCV=0.001`, `r0=1`, `rmax=1,000`, and `iterlim=1,000`).

3.7.9 MID

The maximally informative dimension algorithm (Sharpee et al., 2004) estimates a set of filters $\{\hat{v}_1, \hat{v}_2, \dots, \hat{v}_n\}$ that maximize the mutual information between cell responses and projections of the stimuli on those filters, i.e., $\hat{v}_1, \dots, \hat{v}_n = \arg \max_{\tilde{v}_1, \dots, \tilde{v}_n} I(\{X^T \tilde{v}_1, \dots, X^T \tilde{v}_n\}, Y)$, where X and Y are random variables associated with the stimuli and responses, respectively. An implementation of MID is freely available from Dr. Sharpee’s laboratory web site (<http://cnl-t.salk.edu/Download/>). However, this implementation only allows to estimate one filter, i.e., $n = 1$ in the previous maximization. Since a multi-filter implementation of MID is unavailable, we estimated multiple filters sequentially, by finding the first filter using the one-filter implementation of MID, and then searching in the orthogonal space for the next filter (Sharpee, personal communication). This sequential procedure is guaranteed to converge to the right filters provided the stimuli are Gaussian, either with or without correlations (Sharpee, 2007).

Having estimated a first MID filter with D delays, to estimate a second MID filter, we first build a first set of spatio-temporal stimuli set as follows. If $x(n)$ is the stimulus at time n , then the corresponding first spatio-temporal stimulus is $w(n) = [x(n - (D - 1)), \dots, x(n - 1), x(n)]$, where $[\dots]$ is the concatenation operator. Next, the second spatio-temporal stimuli set is obtained by removing from each stimulus in the first spatio-temporal stimuli set its component along the first MID filter. Then, the second MID filter is estimated by running the single-filter implementation of MID with the second spatio-temporal stimuli set. And, in general, having estimated the n^{th} MID filter, the $n + 1^{\text{th}}$ spatio-temporal stimuli set is constructed by removing from each stimulus in the n^{th} spatio-temporal stimuli set its component along the n^{th} MID filter. Then the $n + 1^{\text{th}}$ MID filter is estimated by running the single filter implementation of MID with the $n + 1^{\text{th}}$ spatio-temporal stimuli set.

To improve the MID estimates, we varied the number of bins and number of iterations in MID estimations. We always started with the default parameters (15 bins and 1,000 iterations) and then tried a few other values of these parameters. But the computational cost of MID is very high (on a personal computer, with a 3GHz processor and 2GB of memory, the estimation of a single MID filter requires between 4 and 8 hours, so a complete MID characterization of a cell with 5 filters requires between 20 and 40 hours) and we could not do an exhaustive exploration of the parameter space for the number of bins and iterations.

3.7.10 nSTC

For Gaussian white noise stimuli, if a cell is selective to a set of relevant dimensions, then the variance of stimuli that elicit spikes (spike-triggered stimuli) along these dimensions should be higher or lower than the variance along non-relevant dimensions. The dimensions with high or low variance correspond to the eigenvectors of the auto-covariance matrix with correspondingly high or low eigenvalues. For Gaussian stimuli, Spike-Triggered Covariance (STC) (de Ruyter van Steveninck & Bialek, 1988) identifies

the relevant dimensions of a cell as the eigenvectors of the autocovariance matrix of the spike-triggered stimuli that correspond to significantly high or low eigenvalues.

Touryan et al. (2005) proposed a modification to STC for natural stimuli. This modification starts by whitening the natural stimuli. Denote by U to the matrix of eigenvectors of the autocovariance of the stimuli (one eigenvector per column), and by λ_i to its eigenvalues. The matrix of normalized eigenvectors is defined as

$$U_n = U \begin{pmatrix} \frac{1}{\sqrt{\lambda_1}} & & 0 \\ & \ddots & \\ 0 & & \frac{1}{\sqrt{\lambda_n}} \end{pmatrix} \quad (3.9)$$

Then the whitened natural images are $X_w = XU_n$. STC for natural images now performs a classical STC analysis on the whitened natural images, obtaining a set of relevant dimensions, V_w (one relevant dimension per column). Finally, the desired relevant dimensions of the cell, V (one relevant dimension per column), are $V = U_n V_w$.

The autocovariance matrix of natural images is nearly singular, so its last eigenvalues ($\lambda_i, i \gg 1$) will be very small and tend to amplify noise. To avoid this effect, a threshold is selected and, for each eigenvalue λ_j less than this threshold, the diagonal value $1/\sqrt{\lambda_j}$ in Equation 3.9 is set to zero. The results reported here correspond to using a threshold such that approximately 35% of the eigenvalues were greater than it, as in Touryan et al. (2005).

3.7.11 rSTA

If the input images were Gaussian white noise, the filter α could be estimated by cross-correlating the responses with the images, $\hat{\alpha} = C_{yw}$ (de Boer & Kuyper, 1968). But, natural images are not Gaussian white noise (Field, 1987; Ruderman & Bialek, 1994; Simoncelli & Olshausen, 2001). Nevertheless, if the cell response is a linear function of its inputs, the filter α can be estimated as $\hat{\alpha} = C_{ww}^{-1}C_{yw}$, where C_{ww} represents the

autocorrelation of the inputs (Theunissen et al., 2001). The autocorrelation matrix, C_{ww} , for natural stimuli is nearly singular. Therefore, its true inverse tends to amplify noise. To avoid this problem, we regularized the autocorrelation matrix using the truncated singular value decomposition (Hansen, 1987) and computed the pseudoinverse (Ben-Israel & Greville, 1980) from this regularized matrix (Smyth et al., 2003). The computation of the truncated singular value decomposition uses a threshold to decide how many singular values to include in the regularized matrix. We selected the optimal threshold using k-fold cross-validation (Efron & Tibshirani, 1993).

3.7.12 Multi-dimensional polynomial predictive model

Given a set of M spatio-temporal filters $\{f_1, \dots, f_M\}$, each with a temporal extent of D delays, to predict the response of a cell at time t , we first computed the dot product between each of the M filters and the concatenation of the images presented at times $t, t-1, \dots, t-(D-1)$, i.e., we computed the vector $v_t(i) = f_i^T[x(t), \dots, x(t-D)]$, $1 \leq i \leq M$. Then, we used this vector as input to a second-order multi-dimensional polynomial (Equation 3.10) to generate the predictions at time t of the simulated (Figures 3.4 and 3.13f) and cortical (Figures 3.6f, and 3.6g) complex cells, or as input to a third-order multi-dimensional polynomial (Equation 3.11) to generate the predictions of the cortical simple cell (Figure 3.8). The coefficients of a polynomial model were estimated by minimizing the mean-squared error between cell responses and the polynomial model predictions.

$$y(t) = k_0 + \sum_{i=1}^M k_1(i)v_t(i) + \sum_{i_1=1}^M \sum_{i_2=i_1}^M k_2(i_1, i_2)v_t(i_1)v_t(i_2) \quad (3.10)$$

$$y(t) = k_0 + \sum_{i=1}^M k_1(i)v_t(i) + \sum_{i_1=1}^M \sum_{i_2=i_1}^M k_2(i_1, i_2)v_t(i_1)v_t(i_2) + \sum_{i_1=1}^M \sum_{i_2=i_1}^M \sum_{i_3=i_2}^M k_3(i_1, i_2, i_3)v_t(i_1)v_t(i_2)v_t(i_3) \quad (3.11)$$

To predict responses from the purely spatial filters of the ePPR models without time interactions in Figure 3.13f (orange bar), we used the spatio-temporal polynomial model in Equation 3.12. Given the spatial filters $\{f_1^d, f_2^d\}$, $0 \leq d \leq 1$ (Figure 3.13a), we first computed the dot products between each of these filters and the image at delay d , i.e., we computed the vectors $v_t^d(i) = (f_i^d)^T x(t-d)$, $0 \leq d \leq 1, 1 \leq i \leq 2$. Then, we used these vectors as inputs to the polynomial model in Equation 3.12.

$$\begin{aligned}
y(t) = k_0 &+ \sum_{i=1}^{M_0} k_1^0(i) v_t^0(i) + \sum_{i=1}^{M_1} k_1^1(i) v_t^1(i) + \sum_{i_1=1}^M \sum_{i_2=i_1}^M k_2^{00}(i_1, i_2) v_t^0(i_1) v_t^0(i_2) \\
&+ \sum_{i_1=1}^M \sum_{i_2=i_1}^M k_2^{01}(i_1, i_2) v_t^0(i_1) v_t^1(i_2) + \sum_{i_1=1}^M \sum_{i_2=i_1}^M k_2^{11}(i_1, i_2) v_t^1(i_1) v_t^1(i_2) \quad (3.12)
\end{aligned}$$

3.7.13 Testing for difference in the predictive power of two models

To compare the predictive power of two models, we used them to predict responses to the 8 testing subsets, and computed Pearson correlation coefficients between these predictions and cell responses. In this way, we constructed two sets of 8 correlation coefficients, one for each model. In most cases, the variability of the correlation coefficients in each set was large, and the error bars (size one standard deviation) of the mean correlation coefficients of the two models intersected. However, it was frequently the case that for each testing subset the correlation coefficient of one of the models was larger than that of the other model. So, to compare the predictive power of the two models, for each testing subset we paired the correlation coefficients of the two models, and used a one-sided Wilcoxon signed-rank test (Hollander & Wolfe, 1999) to check if one element of the pair was significantly larger than the other element.

3.8 Appendices

3.8.1 PPR

3.8.1.1 PPR algorithm

A detailed description of the PPR algorithm appears in Friedman (1984a). For completeness we provide an abridged description here. Equation 3.1 describes the PPR model whose parameters are optimized by minimizing Equation 3.2, as described algorithmically in Listing 1. PPR consists of a forward stepwise procedure followed by a backward stepwise procedure. The parameters of the algorithm are the responses, y , the input images x , the length of the forward model, M_L , the length of the final PPR model, M_0 , and the degree of smoothness of the nonlinear functions, d .

Listing 1 PPR

Require: y, x, M_L, M_0, d

- 1: $(\text{model}, r) \leftarrow \text{FORWARD_STEPWISE}(y, x, M_L, d)$ {Built forward model $\hat{y}(x) = \bar{y} + \sum_{m=0}^{M_L} \beta_m \phi_m(\alpha_m^T x)$ }
 - 2: $\text{model} \leftarrow \text{BACKWARD_STEPWISE}(\text{model}, r, x, M_0, d)$ {Obtained model $\hat{y}(x) = \bar{y} + \sum_{m=0}^{M_0} \beta_m \phi_m(\alpha_m^T x)$ }
 - 3: **return** model
-

PPR forward stepwise procedure

For the forward procedure, an initial M_L -term model of the form given in Equation 3.1 is constructed. An algorithmic description of this procedure is given in Listing 2. It first defines the residuals r as the mean-subtracted responses. Then, it obtains an initial estimate of α by reverse correlation. Next, it fits the first term of the model obtaining the approximation $y = \bar{y} + \beta_1 \phi_1(\alpha_1^T x)$. It next defines the new residuals $r = r - \beta_1 \phi_1(\alpha_1^T x)$ and fits to them the second term of the model. This gives the approximation $y = \bar{y} + \beta_1 \phi_1(\alpha_1^T x) + \beta_2 \phi_2(\alpha_2^T x)$. Continuing in this fashion, it arrive at the forward stepwise estimated model.

Listing 2 PPR: FORWARD_STEPWISE

Require: y, x, M_L, d

- 1: $r \leftarrow y - \bar{y}$
 - 2: **for** $m \in 1$ to M_L **do**
 - 3: $\alpha \leftarrow \text{GET_INITIAL_ALPHA}(r, x)$
 - 4: $(\beta_m, \phi_m, \alpha_m) \leftarrow \text{FIT_NEW_TERM}(r, x, \alpha, d)$ {Find $(\beta_m, \phi_m, \alpha_m)$ minimizing $J = \sum_{i=1}^N (r - \beta_m \phi_m(\alpha_m^T x))^2$ }
 - 5: $r = r - \beta_m \phi_m(\alpha_m^T x)$
 - 6: **end for**{Built model $\hat{y}(x) = \bar{y} + \sum_{m=0}^{M_L} \beta_m \phi_k(\alpha_m^T x)$ }
 - 7: **return** **model** = $[(\beta_1, \phi_1, \alpha_1), \dots, (\beta_{M_L}, \phi_{M_L}, \alpha_{M_L})]$
-

PPR backward stepwise procedure

The forward stepwise procedure is greedy and it is not guaranteed to converge to the optimal M term model. The terms in the model estimated by the forward stepwise procedure do not necessarily appear in decreasing order of importance. To avoid this problem, and estimate a model with the M_0 most important terms, a backward stepwise procedure is used. The strategy is to fit the forward model with a relatively large value of M_L , and then to search for models of size $M_{L-1}, M_{L-2}, \dots, M_0$ with optimal terms. The starting parameter values to search for the M -term model with optimal terms are the M most important terms of the model with $M + 1$ terms. Term importance is measured by $|\beta_m|$ ($1 \leq m \leq M + 1$). The starting parameters values to search for the $(M_L - 1)$ -term model with optimal terms are given by the forward stepwise model.

An algorithmic description of the backward stepwise procedure is given in Listing 3. The procedure repeatedly drops terms from the model until the model contains only M_0 terms. After each term is dropped the residuals are adjusted (according to the contributions of the dropped term), and all the terms of the model are refitted.

Listing 3 PPR: BACKWARD_STEPWISE

Require: **model**, r, x, M_0, d

- 1: **for** $m \in M_L$ **downto** M_0 **do**
 - 2: $((\beta, \phi, \alpha), \text{model}) \leftarrow \text{DROP_LEAST_IMPORTANT_TERM}(\text{model})$
 - 3: $r \leftarrow r + \beta \phi(\alpha^T x)$
 - 4: **model** $\leftarrow \text{REFIT_MODEL}(\text{model}, r, x, d)$
 - 5: **end for**
 - 6: **return** **model**
-

The refit model procedure is described in Listing 4. It starts by refitting the β_m parameters in Equation 3.1, while fixing the remaining ϕ_m and α_m parameters. This is a linear problem, with regressors $\phi_m(\alpha_m^T x)$ and dependent variable $y(x)$, that can be solved using standard linear regression techniques. Then, following the PPR estimation strategy, terms are refitted one at a time. In doing so, a term is removed from the model, residuals are adjusted according to the contribution of the removed term, a new term is fitted to the adjusted residuals, the newly fitted term is added to the model, and the residuals are adjusted again according to the contributions of this term.

Listing 4 PPR: REFIT_MODEL

Require: model, r, x, d

```

1: model ← REFIT_MODEL_BETAS(model, r)
2: for  $m \in 1$  to  $M$  do
3:    $(\beta_m, \phi_m, \alpha_m, \text{model}) \leftarrow \text{REMOVE\_TERM\_FROM\_MODEL}(m, \text{model})$ 
4:    $r \leftarrow r + \beta_m \phi_m(\alpha_m^T x)$ 
5:    $(\beta, \phi, \alpha) \leftarrow \text{FIT\_NEW\_TERM}(r, x, \alpha_m, d)$ 
6:   model ← ADD_TERM_TO_MODEL( $(\beta, \phi, \alpha)$ , model)
7:    $r \leftarrow r - \beta \phi(\alpha^T x)$ 
8: end for
9: return model

```

PPR fit new term procedure

When fitting a new term the sum of square error (SSE) in Equation 3.13 is minimized with respect to the smooth function ϕ , and the projection direction α . This is done iteratively. In each iteration α is first fixed and ϕ is adjusted to minimize the SSE. Then ϕ is fixed and α is adjusted to minimize the SSE. This iteration is repeated until the SSE stops decreasing. Finally, β is calculated and ϕ is normalized. The procedure is described algorithmically in Listing 5.

$$SSE = \sum_{i=1}^N (r_i - \phi(\alpha^T x_i))^2 \quad (3.13)$$

Listing 5 PPR: FIT_NEW_TERM

Require: r, x, α, d

- 1: **repeat**
 - 2: Fix α and find ϕ_{k+1} minimizing $SSE(\alpha, \phi) = \sum_{i=1}^N (r_i - \phi(\alpha^T x_i))^2$ {Smooth the scatter plot $(\alpha^T x_i, r_i)$ using a smoothing spline or Friedman's 'super smoother'.}
 - 3: Fix ϕ_{k+1} and update α along a Gauss-Newton descent direction δ_k of $SSE(\alpha, \phi_{k+1}) : \alpha_{k+1} = \alpha_k + \delta_k$.
 - 4: **until** SSE stops decreasing
 - 5: $\beta \leftarrow \sqrt{E\{\phi^2(\alpha^T x)\}}$
 - 6: $\phi \leftarrow \phi/\beta$
 - 7: **return** (β, ϕ, α)
-

To adjust ϕ , with α fixed, the procedure projects the input images onto α , $p_i = \alpha^T x_i$, builds a scatter plot with the pairs of projections and residuals, (p_i, r_i) , and smooths the scatter plot using a smoothing spline (Green & Silverman, 1994), or Friedman's 'super smoother' (Friedman, 1984b), giving the adjusted ϕ .

With ϕ fixed the SSE in Equation 3.13 can be expressed as the L^2 norm of a nonlinear function h depending on α , as shown in Equation 3.14. Then, finding the value of α minimizing the SSE reduces to a nonlinear least-squares problem, which PPR solves using the Gauss-Newton algorithm (Bertsekas, 1999).

$$SSE = \sum_{i=1}^N (r_i - \phi(\alpha^T x_i))^2 = \sum_{i=1}^N |h_i(\alpha)|^2 = \|h(\alpha)\|_2^2 \quad (3.14)$$

$$\text{where } h_i(\alpha) = r_i - \phi(\alpha^T x_i)$$

$$\text{and } h(\alpha) = (h_1(\alpha), \dots, h_N(\alpha))$$

Briefly, the Gauss-Newton algorithm produces a sequence $\alpha_1, \alpha_2, \dots$ such that $\|h(\alpha_k)\|_2^2$ converges to a minimum. Given α_k , we obtain α_{k+1} using Equation 3.15. The value of δ_k is selected to minimize $\|h(\alpha_{k+1})\|_2^2$. In doing so the Gauss-Newton algorithm approximates $h(\alpha_{k+1})$ using its first-order Taylor expansion, $\hat{h}(\alpha_{k+1}) = h(\alpha_k) + \nabla^T h(\alpha_k) \delta_k$, and estimates δ_k satisfying Equation 3.16. From the first-order necessary conditions for

a minimum of Equation 3.16, δ_k is the solution of the linear system of equations in Equation 3.17.

$$\alpha_{k+1} = \alpha_k + \delta_k \quad (3.15)$$

$$\begin{aligned} \delta_k &= \arg \min_{\delta} \frac{1}{2} \|h(\alpha_k) + \nabla^T h(\alpha_k) \delta\|_2^2 \\ &= \arg \min_{\delta} \frac{1}{2} [\|h(\alpha_k)\|_2^2 + 2\nabla h(\alpha_k)h(\alpha_k)\delta + \delta^T \nabla h(\alpha_k) \nabla^T h(\alpha_k) \delta] \end{aligned} \quad (3.16)$$

$$\nabla h(\alpha_k) \nabla^T h(\alpha_k) \delta_k = -\nabla h(\alpha_k) h(\alpha_k) \quad (3.17)$$

In PPR α_k is not updated until $\|\hat{h}(\alpha_k)\|_2^2$ converges to a minimum. Instead, α_k is updated only once and then the algorithm proceeds with the next iteration.

3.8.1.2 Selection of PPR hyperparameters

Number of terms in the forward (M_L) and final (M_0) model

As described in Section 3.8.1.1, the PPR algorithm is controlled by parameters M_L , the number of terms in the forward model, and M_0 , the number of terms in the final model. To estimate these parameters we fitted a PPR model using a very large value for M_L , usually $M_L = 10$, and with $M_0 = 1$. The PPR algorithm returns the goodness of fit (quantified by the SSE in Equation 3.2 for the training data) of the backfitted models having between M_0 and M_L terms. A plot of these goodness of fit values, as a function of the number of terms in the PPR model, usually has an L shape. Before a given point, as we increase the number of terms in PPR models, the goodness of fit improves considerably. But, after

this point, increasing the number of terms improves the goodness of fit only marginally. So, we selected the number of terms at which the SSE stops decreasing considerably as the number of terms, M_O , for the final PPR model. To allow refitting of the terms in the final PPR model, we set the number of terms in the forward model to $M_L = M_O + 3$.

Degree of smoothness of the nonlinear functions (d)

To estimate the smooth functions we used super smoother (Friedman, 1984b) with automatic span selection.

3.8.2 ePPR

3.8.2.1 ePPR algorithm

Equation 3.6 describes the ePPR model whose parameters are optimized by minimizing Equation 3.7. To overcome the curse of dimensionality, ePPR retains the optimization strategy of PPR. There are three main differences between the PPR and ePPR estimation algorithms. First, ePPR extends PPR to become spatio-temporal. For the model without time interactions (Section 3.2), after fitting the responses to images presented at the same time bin as the responses, residuals are fitted to images presented in previous time bins. For models with time interaction, images at several delays are concatenated to form spatio-temporal inputs, and spatial ePPR models are then estimated using these spatio-temporal inputs, i.e., the ePPR estimation procedure is invoked with a delay $D^L = 0$ for the forward model. Second, to avoid problems caused by correlations in natural images, ePPR uses a Trust Region method, instead of a Gauss Newton method, to solve the nonlinear least-squares problem in Equation 3.19. Third, to obtain smooth projection directions, ePPR penalizes the criterion in Equation 3.3 used in PPR to fit a new term, as shown in Equation 3.18. The first difference requires minor changes in the FORWARD_STEPWISE, BACKWARD_STEPWISE, and REFIT_MODEL procedures for the estimation of models without time interactions, and requires no change in the PPR algorithm for

the estimation of models with time interaction. The last two differences only require changing the FIT_NEW_TERM procedure.

The ePPR estimation algorithm is described in Listing 6. It consists of a forward stepwise procedure followed by a backward stepwise procedure, and a model selection procedure. The parameters of the algorithm are the responses, y , the input images, x , the number of delays for the forward model, D^L , the number of terms at each delay d for the forward model, M_d^L , the regularization parameter for the α 's, λ , and the number of degrees of freedom for the ϕ 's, d .

Listing 6 ePPR

Require: $y, x, D^L, \{M_0^L, \dots, M_{D^L}^L\}, \lambda, d$

- 1: $(\text{model}, r) \leftarrow \text{FORWARD_STEPWISE}(y, x, D^L, \{M_0^L, \dots, M_{D^L}^L\}, \lambda, d)$
 {Built forward model $\hat{y}(i) = \bar{y} + \sum_{d=0}^{D^L} \sum_{m=0}^{M_d^L} \beta_{m,d} \phi_{m,d}(\alpha_{m,d}^T x_{i-d})$ }
- 2: $\text{models} \leftarrow \text{BACKWARD_STEPWISE}(r, x, \lambda, d)$
 {Obtained models of the form $\hat{y}(x) = \bar{y} + \sum_{d=0}^D \sum_{m=0}^{M_d} \beta_{m,d} \phi_{m,d}(\alpha_{m,d}^T x_{i-d})$, having between 1 term and the maximum number of terms in the forward model.}
- 3: $\text{model} \leftarrow \text{SELECT_BEST_MODEL}(\text{models})$
- 4: **return** model

ePPR forward stepwise procedure

The forward procedure estimates an ePPR model with delays $d = 0, \dots, D^L$ and containing M_d^L terms at delay d . An algorithmic description of this procedure is given in Listing 7. It first defines the residuals r as the mean-subtracted responses. Then it fits to these residuals M_0^L terms operating on images presented at the same time bin as the responses. These M_0^L terms are fitted with the ePPR FIT_NEW_TERM procedure. Next, the input images are shifted in time, so that the image presented at time $i - 1$ is displaced to time i , and a new set of M_1^L terms operating on these shifted images is fitted to the response residuals. In this way the response at time i has been approximated by M_0^L terms operating on the image presented at time i , plus additional M_1^L terms operating on the image presented at time $i - 1$. After shifting the image D^L times, the forward ePPR model is constructed.

Listing 7 ePPR: FORWARD_STEPWISE

Require: $y, x, D^L, \{M_0^L, \dots, M_{D^L}^L\}, \lambda, d$

```
1:  $r \leftarrow y - \bar{y}$ 
2: for  $d \in 0$  to  $D^L$  do
3:   for  $m \in 1$  to  $M_d^L$  do
4:      $\alpha \leftarrow \text{GET\_INITIAL\_ALPHA}(r, x)$ 
5:      $(\beta_{m,d}, \phi_{m,d}, \alpha_{m,d}) \leftarrow \text{FIT\_NEW\_TERM}(r, x, \lambda, d)$ 
6:      $r = r - \beta_{m,d} \phi_{m,d}(\alpha_{m,d}^T x)$ 
7:   end for
8:    $x_i \leftarrow x_{i-1}$ 
9: end for {Built model  $\hat{y}(i) = \bar{y} + \sum_{d=0}^{D^L} \sum_{m=1}^{M_d^L} \beta_{m,d} \phi_{m,d}(\alpha_{m,d}^T x_{i-d})$ }
10: return model =  $[(\beta_{1,0}, \phi_{1,0}, \alpha_{1,0}), \dots, (\beta_{M_{D^L}^L, D^L}, \phi_{M_{D^L}^L, D^L}, \alpha_{M_{D^L}^L, D^L})]$ 
```

ePPR backward stepwise procedure

The ePPR backward stepwise procedure builds a list containing models having between 1 and $\sum_{d=0}^{D^L} M_d^L$ terms. The procedure is described in Listing 8. It is similar to the same procedure in PPR (Listing 3). The input **model** for this procedure is the output of the ePPR FORWARD_STEPWISE procedure. The procedure operates iteratively. Suppose that at the beginning of one iteration **model** contains the M terms that best fit the response, then **model** is saved in the **models** list, the least important term is dropped from **model**, the residuals are adjusted according to the contribution of the dropped term, and the parameters of the dropped **model** are refitted to the adjusted residuals. Then, by the end of the iteration **model** approximates a model with $M - 1$ terms that best fit the response. In the next iteration **model** will be saved in the list of **models** and a model with the best $M - 2$ terms fitting the response will be estimated. By the end of the loop, the list **models** will contain $\sum_{d=0}^{D^L} M_d^L$ models, where the M^{th} model in this list will be the M -term model that best approximates the response in the L^2 -norm.

The ePPR REFIT_MODEL procedure is described in Listing 9. It is similar to the same procedure in PPR (Listing 4), but adds a few statements to refit terms using appropriately delayed images. The procedure begins by refitting the β parameters. Then it enters a loop where each term is removed from the model, the images are shifted according to the delay of the removed term, residuals are adjusted according to the contribution of the

Listing 8 ePPR: BACKWARD_STEPWISE

Require: $\text{model}, r, x, \lambda, d$

```
1:  $\text{models} \leftarrow []$ 
2: for  $m = \text{nTerms}(\text{model})$  downto 2 do
3:    $\text{models} \leftarrow [\text{models}, \text{model}]$ 
4:    $((\beta, \phi, \alpha), \text{model}) \leftarrow \text{DROP\_LEAST\_IMPORTANT\_TERM}(\text{model})$ 
5:    $r \leftarrow r + \beta \phi(\alpha^T x)$ 
6:    $\text{model} \leftarrow \text{REFIT\_MODEL}(\text{model}, r, x, \lambda, d)$ 
7: end for
8:  $\text{models} \leftarrow [\text{model}, \text{models}]$ 
9: return  $\text{models}$ 
```

removed term, a new term is fitted to the adjusted residuals, this new term is added to the back of the model, and the residuals are adjusted according to the contribution of the new term.

Listing 9 ePPR: REFIT_MODEL

Require: $\text{model}, r, x, \lambda, d$

```
1:  $\text{model} \leftarrow \text{REFIT\_MODEL\_BETAS}(\text{model})$ 
2:  $\text{noTerms} \leftarrow \text{noTerms}(\text{model})$ 
3: for  $m \in 1$  to  $\text{noTerms}$  do
4:    $(\beta_m, \phi_m, \alpha_m, \text{delay}, \text{model}) \leftarrow \text{REMOVE\_FIRST\_TERM\_FROM\_MODEL}(m, \text{model})$ 
5:    $xs_i \leftarrow x_{i-\text{delay}}$ 
6:    $r \leftarrow r + \beta_m \phi_m(\alpha_m^T xs)$ 
7:    $(\beta, \phi, \alpha) \leftarrow \text{FIT\_NEW\_TERM}(r, xs, \alpha_m, \lambda, d)$ 
8:    $\text{model} \leftarrow \text{ADD\_TERM\_TO\_BACK\_OF\_MODEL}((\beta, \phi, \alpha), \text{model})$ 
9:    $r \leftarrow r - \beta \phi(\alpha^T xs)$ 
10: end for
11: return  $\text{model}$ 
```

ePPR fit new term procedure

To avoid estimates that overfit noise in the responses, the ePPR estimation algorithm penalizes estimates, \hat{y} , containing non-smooth projection directions. This is accomplished by adding penalty terms to the PPR estimation criterion in Equation 3.2, as shown in Equation 3.7. To account for these penalty terms the objective function used in PPR to estimate a new term, Equation 3.3, is expanded to that used in ePPR, Equation 3.18. The ePPR new term procedure optimizes the criterion J in Equation 3.18 with respect

to the smooth function ϕ , and the projection direction α . This is performed iteratively. In each iteration α is first fixed and ϕ is adjusted to minimize J . Then ϕ is fixed and α is adjusted to minimize J . This iteration is repeated until the reduction in J for two consecutive iterations falls below a convergence value. Finally, β is calculated and ϕ is normalized. The procedure is described algorithmically in Listing 10.

$$J = \sum_{i=1}^N (r_i - \phi(\alpha^T x_i))^2 + \lambda \|L\alpha\|^2 \quad (3.18)$$

Listing 10 ePPR: FIT_NEW_TERM

Require: r, x, α, λ, d

- 1: **repeat**
 - 2: Fix α and find ϕ_{k+1} minimizing $\tilde{J}(\alpha, \phi) = \sum_{i=1}^N (r_i - \phi(\alpha^T x_i))^2$ {Smooth the scatterplot $(\alpha^T x_i, r_i)$ using a smoothing spline with d degrees of freedom}
 - 3: Fix ϕ_{k+1} and update α along a Trust Region descent direction δ_k of $J(\alpha, \phi_{k+1}) = \sum_{i=1}^N (r_i - \phi(\alpha^T x_i))^2 + \lambda \|L\alpha\|^2$: $\alpha_{k+1} = \alpha_k + \delta_k$.
 - 4: **until** J stops decreasing
 - 5: $\beta \leftarrow \sqrt{E\{\phi^2(\alpha^T x)\}}$
 - 6: $\phi \leftarrow \phi/\beta$
 - 7: **return** (β, ϕ, α)
-

Note that when α is fixed the second term in Equation 3.18 is a constant. So to adjust ϕ , with α fixed, it suffices to minimize the first term in Equation 3.18. This is done in the same way as in PPR. The procedure projects the input images onto α , $p_i = \alpha^T x_i$, builds a scatter plot with the pairs of projections and residuals, (p_i, r_i) , and sets ϕ as the smoothing spline that best approximates, in the L^2 norm, the points in the scatter plot. The degrees of freedom d , controls the smoothness of the estimated splines. Appendix 3.8.2.2 describes the procedure we used to select this parameter.

With ϕ fixed, Equation 3.18 can be expressed as the L^2 norm of a nonlinear function \tilde{h} depending on α , as shown in Equation 3.19. Then, finding the value of α that minimize Equation 3.19 for a fixed ϕ reduces to a nonlinear least-squares problem. For the

reasons explained in Section 3.2.3, in ePPR this problem is solved using the Trust Region method (Nocedal & Wright, 2006).

$$\begin{aligned}
J &= \sum_{i=1}^N (r_i - \phi(\alpha^T x_i))^2 + \lambda \|L\alpha\|^2 = \sum_{i=1}^N |h_i(\alpha)|^2 + \sum_{j=1}^m |\sqrt{\lambda} L\alpha_j|^2 = \|\tilde{h}(\alpha)\|_2^2 \quad (3.19) \\
\text{where } h_i(\alpha) &= r_i - \phi(\alpha^T x_i) \\
\text{and } \tilde{h}(\alpha) &= (h_1(\alpha), \dots, h_N(\alpha), \sqrt{\lambda}(L\alpha)_1, \dots, \sqrt{\lambda}(L\alpha)_m)
\end{aligned}$$

As in PPR, α_k is not updated until $\|\hat{h}(\alpha_k)\|_2^2$ converges to a minimum. Instead, α_k is updated only once and then the algorithm proceeds with the next iteration.

ePPR select best model procedure

The backward-stepwise procedure (Listing 8) returns a set of models having between 1 and $\sum_{d=0}^{D^L} M_d^L$ terms. The final ePPR estimate is selected from this set. Several strategies have been proposed for selecting the best model from a set of candidate models (Burnham & Anderson, 2002). Here we used cross-validation, as described in Section 3.7.6.

3.8.2.2 Selection of ePPR hyperparameters

Below we describe the ePPR hyperparameters and the procedure we used to select their values. The hyperparameters selected to estimate the ePPR models used in this paper are shown in Section 3.7.8.

Number of delays (D^L) and number of terms per delay (M_d^L) for the forward model

ePPR builds a forward model with a maximum delay of D^L and with M_d^L terms at each delay d , $0 \leq d \leq D^L$. We set the parameters D^L and M_d^L so that the forward model contained enough terms per delay, and enough delays, to characterize the cell response, as indicated below.

Starting from the forward model, the backward stepwise procedure returns a list of models where the n^{th} model in this list is the ePPR model with n terms that best predicts the cell responses. Then, the model selection procedure selects from this list the final ePPR as that model with the minimum number of terms that best predicts the cell responses. We set the values of D^L and M_d^L large enough so that the model selected by the model selection procedure contains at most $M_d^L - 1$ terms at every delay d and the maximum delay is at most $D^L - 1$. This guarantees that at least 1 term at every delay, and all the terms at delay D^L , were irrelevant for the model maximizing predictive power. Thus, the forward model contained more terms per delay, and more delays, than were needed by the model maximizing predictive power.

Regularization parameters for the filters (λ)

To avoid estimates overfitting noise in the responses we penalized estimates having non-smooth projection directions, as show in Section 3.2.3, by adding penalty terms to the ePPR optimization criterion in Equation 3.7. Each of these penalty terms has the form $\lambda \|L\alpha_{m,d}\|^2$. We chose λ by cross validation. We estimated ePPR models with different values of λ and chose the value of λ maximizing predictions to data not used in fitting the model parameters.

Degrees of freedom for the spines (d)

The degree of smoothness of the splines used to fit the nonlinear functions $\phi_{m,d}$ is controlled by the degrees of freedom parameter, d . The standard procedure is to estimate the value of this parameter from training data using standard or generalized cross validation (Green & Silverman, 1994). However, for the large levels of noise in neural responses, these methods performed poorly. Better results were obtained by choosing d by cross-validation. For the models estimated with intermediate level of noise for the simulated cell, and for the models estimated with the largest amount of data for the cortical cells, we verified, using cross-validation, that setting $d = 5$ provided reasonable results. So, to

reduce the number of model estimates, for all estimations we fixed the degrees of freedom for the splines to $d = 5$.

Convergence values for fitting new terms (`addTermsCV` and `refitCV`)

ePPR fits a new term by repeatedly minimizing the criterion J in Equation 3.18, first with respect to α and then with respect to ϕ . These minimizations continue until the reduction in J for two consecutive iterations falls below a convergence value. The ePPR algorithm provides two convergence value hyperparameters `addTermsCV` and `refitCV`, the former one is used when adding a new ePPR to the forward model, and the latter one is used when refitting an ePPR term. For all the models estimated in this article we used `addTermsCV=0.01` and `refitCV=0.001`.

Trust region hyperparameters (`r0`, `rmax`, and `iterlim`)

At each iteration, the Trust Region algorithm minimizes a target function on a restricted ‘trust’ region where the function behaves well. The size of this trust region is changed adaptively through the minimization (Nocedal & Wright, 2006). The `r0` and `rmax` hyperparameters give the initial and maximal size of the trust region. The hyperparameters `iterlim` controls the maximum number of iterations in the Trust Region algorithm. For all the models estimated in this article we used `r0=1`, `rmax=1,000`, and `iterlim=1,000`.

3.8.3 Proofs

Proposition 1. *Let $x \in \mathbb{R}^p$ and $f(x)$ be a polynomial. Then*

$$f(x) = \bar{y} + \sum_{m=1}^{M_0} \beta_m \phi_m(\alpha_m^T x) \quad (3.20)$$

for constants \bar{y} , β_m , vectors $\alpha_m \in \mathbb{R}^p$, and univariate functions ϕ_m .

Proof. Any polynomial $f(x)$ can be written in the form $f(x) = \sum_{i=0}^n f_m(x)$ where $f_m(x)$ is an homogeneous polynomial of degree m^5 . Proposition 2 proves that any homogeneous polynomials $f_m(x)$ can be written in the form of Equation 3.20. Then, because the sum of expressions of the form of Equation 3.20 is another expression of the form of Equation 3.20, the proof is complete. □

Proposition 2 is a more detailed and simplified version of a result by Diaconis and Sahshahani (1984, Proposition 1). We provide this proof here for completeness, and because the proof is very elegant. Our contribution to the result from Diaconis and Sahshahani (1984) appears in Lemma 2, that replaces the algebraic independence argument, at the end of the original proof, by a simpler explanation using only concepts from linear algebra.

Proposition 2. *Let $r = \binom{m+p-1}{m}$ be the number of distinct monomials of degree m . Then there exist r distinct directions a^1, a^2, \dots, a^r in \mathbb{R}^p such that any homogeneous polynomial f of degree m can be written as $f(x) = \sum_{j=1}^r \alpha_j (x^T a^j)^m$ for some real numbers α_j .*

Proof. The space of homogeneous polynomial of degree m is an r -dimensional vector space over the real numbers. Due to this dimensionality, to prove the proposition it suffices to show that we can find r directions $\{a^j\}_{j=1}^r$, such that the polynomials $\{(x^T a^j)^m\}_{j=1}^r$ are linearly independent.

Let $m_i(x)$, $1 \leq i \leq r$, be an enumeration of all the monomials in homogeneous polynomials of degree m . For each monomial $m_i(x)$, let D_i be the associated differential operator (e.g., if $m_i(x) = x_1^2 x_2 x_3$, $D_i = \partial^4 / \partial^2 x_1 \partial x_2 \partial x_3$). Lemma 1 proves that $D_i(x^T a^j)^m = m! m_i(a^j)$.

Suppose that for any $\{a^j\}_{j=1}^r$ the polynomials $\{(x^T a^j)^m\}_{j=1}^r$ were linearly dependent. This happens if and only if, for any $\{a^j\}_{j=1}^r$, the equation $\sum_{j=1}^r c_j (x^T a^j)^m = 0$ admits

⁵An homogeneous polynomial of degree m is a polynomial whose monomials with nonzero coefficients all have the same total degree m . For example, $x_1^7 + x_1 x_2^4 x_3^2 + x_2^2 x_3^5$ is an homogeneous polynomial of degree $m = 7$; the sum of the exponents in each term is always 7.

a non-trivial solution. Then, applying D_i to both sides of the previous equality we get that, for any $\{a^j\}_{j=1}^r$, the system of equations $\sum_{j=1}^r m_i(a^j) c_j = 0$, $1 \leq i \leq r$, admits a non-trivial solution. Then, for any $\{a_l^j\}_{j=1}^r$, the determinant of the matrix associated with this system of equations must be zero, i.e., $\det(A) = 0$, with $A_{i,j} = m_i(a_j)$. As a function of the coefficients of the directions, $\{a_l^j\}$, $1 \leq j \leq r$, $1 \leq l \leq p$, $\det(A)$ is a non-zero polynomial with coefficients in the field of the rationals. Thus, the evaluation of the non-zero polynomial $\det(A)$ should equal zero for any possible assignment of directions $\{a^j\}_{j=1}^r$. But this is not possible according to the contrapositive of Lemma 2⁶. This contradiction arose because we supposed that that for any $\{a^j\}_{j=1}^r$ the polynomials $\{(x^T a^j)^m\}_{j=1}^r$ were linearly dependent. Therefore, there must exist a set of directions $\{a^j\}_{j=1}^r$ such that the polynomials $\{(x^T a^j)^m\}_{j=1}^r$ are linearly independent, and the proposition is proved. \square

Lemma 1. *Let $x \in \mathbb{R}^p$ and $m_i(x) = x_1^{n_1} x_2^{n_2} \dots x_p^{n_p}$, with $m = \sum_{j=1}^p n_j$. Associate with $m_i(x)$ the differential operator $D_i = \frac{\partial^m}{\partial x_1^{n_1} \partial x_2^{n_2} \dots \partial x_p^{n_p}}$. Then $D_i(x^T a^j)^m = m! m_i(a^j)$.*

Proof. By the multinomial theorem

$$(x^T a^j)^m = \sum_{k_1, k_2, \dots, k_p} \frac{m!}{k_1! k_2! \dots k_p!} (x_1 a_1^j)^{k_1} (x_2 a_2^j)^{k_2} \dots (x_p a_p^j)^{k_p} \quad (3.21)$$

where the summation is taken over all non-negative integers k_1 through k_p such that $\sum_{j=1}^p k_j = m$. Applying D_i to both sides of Equation 3.21 we obtain

$$D_i(x^T a^j)^m = \sum_{k_1, k_2, \dots, k_p} m! (a_1^j)^{k_1} (a_2^j)^{k_2} \dots (a_p^j)^{k_p} D_i \frac{x_1^{k_1} x_2^{k_2} \dots x_p^{k_p}}{k_1! k_2! \dots k_p!}$$

⁶The contrapositive of Lemma 2 says that if F is a non-zero polynomial, with n variables and coefficients in a infinite field k , then exist $\{b_j\}_{j=1}^n \subset \mathbb{R}$ such that $F(b_1, b_2, \dots, b_n)$ is not zero. Then, because $\det(A)$ is a non-zero polynomial with rp variables and coefficients in the infinite field of the rationals, there must exists $\{a_l^j\} \subset \mathbb{R}$, $1 \leq j \leq r$, $1 \leq l \leq p$ such that $\det(A) \neq 0$.

But

$$D_i \frac{x_1^{k_1} x_2^{k_2} \dots x_p^{k_p}}{k_1! k_2! \dots k_p!} = \begin{cases} 1 & \text{if } k_1 = n_1, k_2 = n_2, \dots, k_p = n_p \\ 0 & \text{otherwise} \end{cases}$$

So

$$D_i(x^T a^j)^m = m! (a_1^j)^{n_1} (a_2^j)^{n_2} \dots (a_p^j)^{n_p} = m! m_i(a^j)$$

□

Lemma 2. *Let k be an infinite field, and F a polynomial with n variables and coefficients in k . If $F(b_1, b_2, \dots, b_n) = 0$, for all $\{b_j\}_{j=1}^n \subset \mathbb{R}$. Then $F = 0$.*

Proof. By induction on n .

$n = 1$: a non-zero and one-dimensional polynomial F of degree m with coefficients on an infinite field has at most m roots. Then, if $F(b_1) = 0$ for all $b_1 \in \mathbb{R}$, it follows that $F = 0$.

$n \rightarrow n + 1$: if F is a polynomial of degree m with coefficients in k and variables x_1, \dots, x_n, x_{n+1} , then we can write $F = \sum_{i=0}^m F_i x_{n+1}^i$, where F_i is a polynomial with coefficients in k and variables x_1, \dots, x_n . For example, if $n = 1$ and $m = 2$, $F = c_0 + c_{11}x_1 + c_{12}x_2 + c_{211}x_1^2 + c_{212}x_1x_2 + c_{222}x_2^2 = (c_0 + c_{11}x_1 + c_{211}x_1^2) + (c_{12} + c_{212}x_1)x_2 + c_{222}x_2^2 = F_0 + F_1x_2 + F_2x_2^2$, with $F_0 = c_0 + c_{11}x_1 + c_{211}x_1^2$, $F_1 = c_{12} + c_{212}x_1$, and $F_2 = c_{222}$.

Suppose $F(b_1, b_2, \dots, b_{n+1}) = 0$, for all $b_1, b_2, \dots, b_{n+1} \in \mathbb{R}$. Take $\tilde{b}_1, \tilde{b}_2, \dots, \tilde{b}_n \in \mathbb{R}$ and define $G(x_{n+1}) = F(\tilde{b}_1, \tilde{b}_2, \dots, \tilde{b}_n, x_{n+1})$. Then $G(b_{n+1}) = F(\tilde{b}_1, \tilde{b}_2, \dots, \tilde{b}_n, b_{n+1}) = 0$ for all $b_{n+1} \in \mathbb{R}$. So, by the inductive hypothesis, $G = 0$, or $F_i(\tilde{b}_1, \tilde{b}_2, \dots, \tilde{b}_n) = 0$, for $0 \leq i \leq m$. Because $\tilde{b}_1, \tilde{b}_2, \dots, \tilde{b}_n$ are arbitrary elements in \mathbb{R} , we have that $F_i(b_1, b_2, \dots, b_n) = 0$, for $0 \leq i \leq m$ and for all $b_1, b_2, \dots, b_n \in \mathbb{R}$. So by the inductive hypothesis $F_i = 0$. Thus, $F = \sum_{i=0}^m F_i x_{n+1}^i = 0$.

□

3.8.4 ePPR models without time interactions

Figures 3.10a and 3.10b show the parameters of an example ePPR model without time interaction estimated from responses to natural stimuli, and with intermediate amount of noise (0.56 spikes/image), of the simulated complex cell (Equation 3.8). Nonlinear interactions between pixels of images at different delays are relevant to characterize the responses of the simulated complex cell (Section 3.3), but these nonlinear interactions cannot be accounted by ePPR models without time interaction (Section 3.2.2). Despite this, the estimated filters (Figure 3.10a) well approximate the true filters of the simulated model (Figure 3.2a). Also, the estimated nonlinear functions (Figure 3.10b) correctly recovered the facilitatory/suppressive nature of the associated filters. Figure 3.10c plots the predictions of ePPR models with (red curve) and without (pink curve) time interaction. For comparison, it re-plots, from Figure 3.4, the predictions of second-order multi-dimensional polynomials using ePPR filters with time interactions (orange curve). At low and intermediate noise levels, 5.62 and 0.56 spikes/image, respectively, predictions of the ePPR models with time interaction were significantly better than those of the polynomial models (Wilcoxon signed-rank test, $p < 0.01$, Section 3.7.13), and at all noise levels predictions of the ePPR model with time interactions were significantly better than those ePPR models without time interaction ($p < 0.01$).

The parameters of an example ePPR models without time interactions estimated from responses to natural stimuli of the cortical complex cell (Section 3.4) are shown in Figures 3.11a and 3.11b. At each delay, the filters without time interaction match well the corresponding frames of the filters with time interaction (Figure 3.6a). For instance, the filters at delay 0 of the model without time interaction (first row in Figure 3.11a), match well the first frames of the three most important filters of the model with time interaction (first frames of the three leftmost filters in Figure 3.6a). The nonlinear function of the model without time interaction correspond well with those of the model with time interaction. For example, all the nonlinear functions at delays 0 and 1 of the model without time interaction are facilitatory, and those at delay 2 are suppressive (Figure 3.11b). In

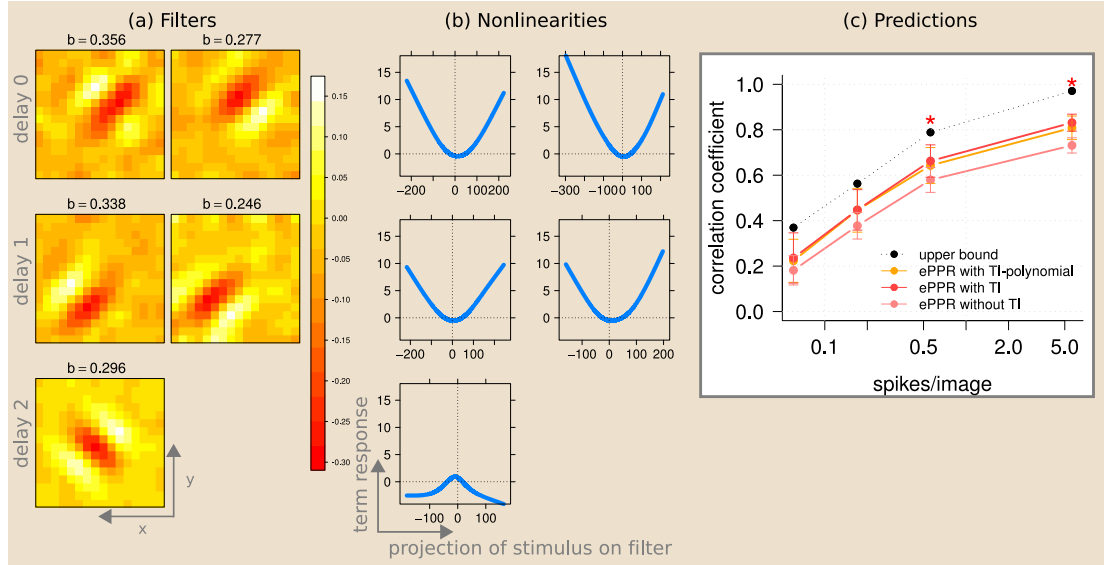


Figure 3.10: Simulated cell: ePPR models without time interaction estimated from responses to natural stimuli. (a,b): filters (a) and nonlinear functions (b) of an example model estimated from responses to natural stimuli with intermediate amount of noise (0.56 spikes/image). The titles in (a) are the corresponding β coefficients. (c) predictive power of ePPR models with and without time interaction compared to that of a polynomial model. Orange curve: predictions for a second-order multi-dimensional polynomial constructed with ePPR filters with time interaction (re-plotted from Figure 3.4a). Red curve: predictions from ePPR models with time interaction. Pink curve: predictions from ePPR models without time interaction. Black curve: upper bound on correlation coefficients. Light red asterisks mark number of spikes/image at which predictions of the ePPR models with time interaction were significantly better than those of the polynomial models. Despite the mismatch between the simulated model in Equation 3.8, that incorporates time interactions between pixels of images at different delays, and the ePPR model without time interactions, that cannot model these interactions, the estimated filters (Figures 3.10a) well approximate the true filters (Figure 3.2a), and the estimated nonlinear functions correctly recovered the facilitatory/suppressive nature of the associated filters.

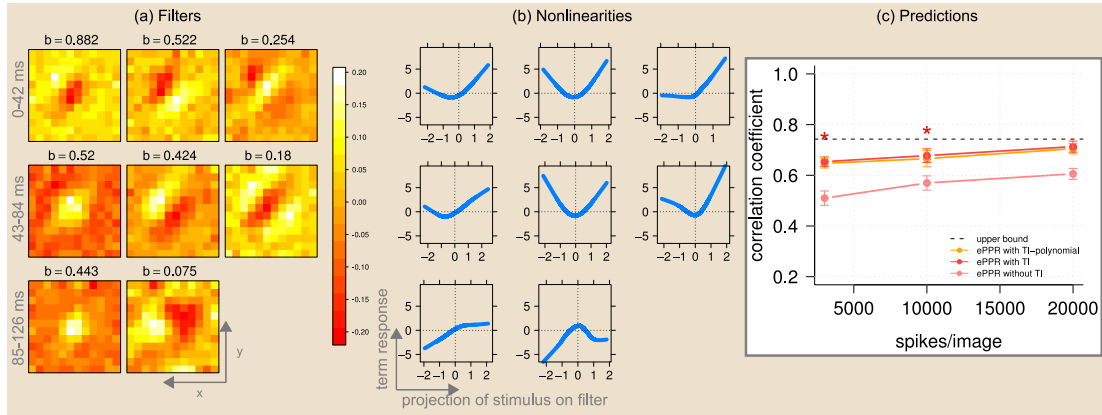


Figure 3.11: Complex cell: ePPR models without time interaction estimated from responses to natural stimuli. Same format at Figure 3.10. The filters and nonlinear function of this model are similar to those of the model with time interaction (Figure 3.6). The ePPR model with time interaction predicts significantly better than the ePPR without time interactions, demonstrating the relevance of nonlinear interactions between pixels of images at different delay for the response of this complex cell.

agreement, the nonlinear functions of the four most important terms of the model with time interaction, whose filters have most structure at delays 0 and 1, are facilitatory, and the nonlinear function of the least important term of the natural model with time interaction, whose filter has most structure at delay 2, is suppressive (Figure 3.6b). As for the simulated cell, Figure 3.11c plots the predictions of ePPR models with (red curve) and without (pink curve) time interaction, as well as those of a polynomial model using the filters of ePPR models with time interaction (orange curve). ePPR models with time interaction predict significantly better than ePPR models without time interaction. This shows that nonlinear interactions between pixels of images at different delays are relevant to predict the responses of this complex cell. Also, ePPR models with time interaction yield better or equal predictions than polynomial models.

The ePPR model without time interaction estimated from responses of the simulated LNL cell is described in Section 3.3.3, and that estimated from responses of the simulated complex cell with temporally correlated inputs is described in Appendix 3.8.6.

3.8.5 Varying the amount of divisive inhibition

Due to the divisive normalization, the simulated model in Equation 3.8 cannot be represented exactly by the ePPR model in Equation 3.6. However, as shown in Figure 3.2, ePPR produced good approximations. To check if this positive result only holds for the particular amount of inhibition used in our simulations, we estimated ePPR models from responses with different amounts of inhibition.

We call the denominator in Equation 3.8 the inhibitory factor. The solid line in Figure 3.12a, re-plotted from Figure 3.2d, shows the principal angles between the true filters and those of ePPR models estimated from simulated responses with a mean inhibitory factor of 4.26, i.e., where the denominator in Equation 3.8 reduced the numerator, on the average, 4.26 times. The dashed and dotted lines plot the principal angles for ePPR models estimated from simulated responses with a mean inhibitory factor of 20.58 and 40.17, respectively.

When the mean inhibitory factor is increased by a factor of 5, from 4.26 to 20.59, only the first principal angle increases marginally. And when the mean inhibitory factor is increased by a factor of 10, from 4.26 to 40.17, only the third principal angle increases substantially. This happens because two of the five ePPR models estimated from responses with a mean inhibitory factor of 40.17 did not recover the inhibitory filter⁷. To visualize the impact of increasing the amount of inhibition on ePPR estimates, Figures 3.12b and 3.12c show the filters and nonlinear functions of the model estimated from responses with the largest inhibition and whose filters were most different from the true filters (according to their principal angles). These estimates capture the most important features of the true excitatory filters in Figure 3.2a.

⁷The ePPR estimation algorithm returns a collection of models having between one term and the number of terms of the forward model (Appendix 3.8.2). From this collection we select the optimal model using a cross-validation procedure (Section 3.7.6). Because in order to compute three principal angles we need three estimated filters, for the two ePPR estimations where the optimal model contained two terms, we used the suboptimal model with three terms returned by the ePPR algorithm to compute the three principal angles.

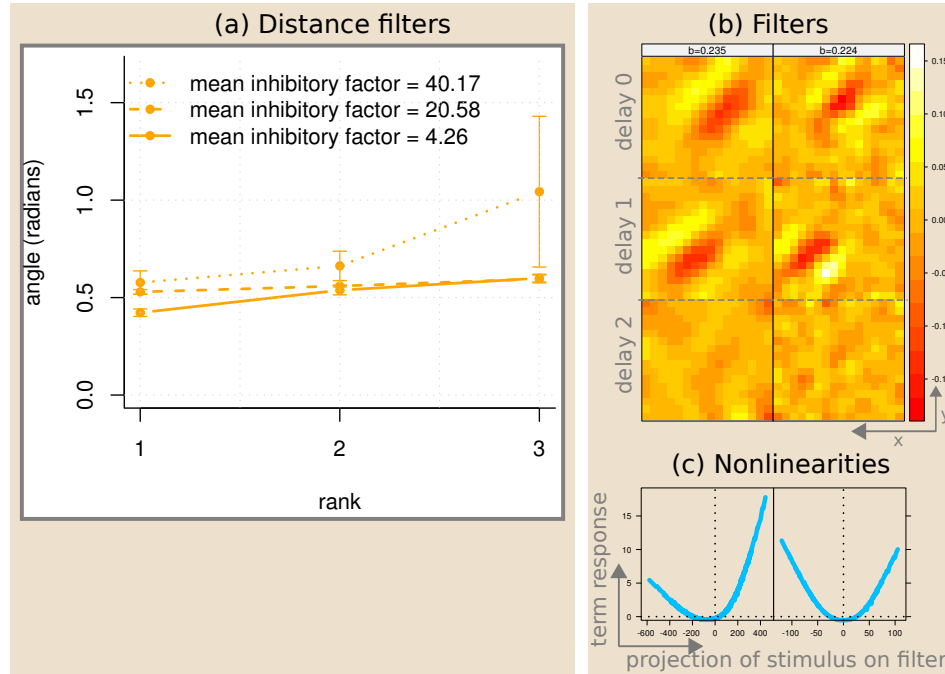


Figure 3.12: Varying the amount of divisive inhibition. (a): principal angles between the true filters of the simulated model (Figure 3.2a) and those of ePPR models with time interaction estimated from responses with varying amount of inhibition. (b,c): filters (b) and nonlinear functions (c) of the ePPR model estimated from responses with the largest inhibition and whose filters were most different from the true filters. Even though ePPR models cannot represent exactly the simulated model with divisive inhibition, they yields good approximations for a broad range of inhibition strengths.

Note that as the mean inhibitory factor increased from 4.26 to 20.58 to 40.17, the mean of the simulated responses (Equation 3.8) decreased from 0.56 to 0.33 to 0.26 spikes per frame. Then, because the noise is Poisson, as inhibition increased the signal to noise ratio in the responses decreased. Hence, the degradation in the quality of the ePPR estimates as the strength of the inhibition is increased does not only reflect the mismatch between the simulated and ePPR models, but is also due to an increasing noise level.

Overall, Figure 3.12 shows that, although ePPR models cannot represent exactly the simulated model with divisive inhibition, they yield good approximations for a large range of inhibition strengths.

3.8.6 Temporally correlated inputs

The simulated models and the real simple and complex cells were probed with stimuli with natural spatial statistics, but that were temporally uncorrelated. To study the effect of temporal correlations on ePPR estimates, we estimated ePPR models from responses of the simulated model with intermediate amount of noise (Section 3.7.3) to the natural movie ensemble (Section 3.7.2).

Figure 3.13a shows the filters of an estimated ePPR model without time interactions. These filters capture the Gabor shape, orientation, and spatial frequency of the true filters of the simulated model (Figure 3.2a). However, we see a few problems in these filters. First, the cross oriented filter at delay 1 (second row and right column in Figure 3.13a) should appear at delay 2. Second, one filter is missing at delay 1. Third, the shapes of the the excitatory filters are distorted. For example, the estimated filters at delay 0 combine features of the true filters at delay 0 and 1. Despite these problems, the estimated nonlinear functions have the correct quadratic shape and properly capture the facilitatory/suppressive nature of the corresponding filters (Figure 3.13b). The estimated ePPR model with time interactions contains only one term. Its filter (Figure 3.13c) captures some features of the excitatory true filters (Figure 3.2a), like orientation, spatial frequency, and position, but its shape is distorted. Its nonlinear function (Figure 3.13d) is excitatory, but does not have the quadratic shape associated with the excitatory filters.

We see that temporal correlations in the inputs degrades the quality of the ePPR estimates. To check if this negative impact of temporal correlations was a problem of ePPR only, or a problem shared by other methods, we estimated MID filters (Section 3.7.9) from the same responses to the natural movie ensemble. Figure 3.13e shows the first three MID filters. Only the first MID filter contains structure. This filter has a Gabor shape, with correct orientation and spatial frequency. However, the position of this filters at delays 0 and 1 is incorrect, and the filter at delay 2 is spurious. So, correlations in the inputs have a negative impact on the estimation of MID filters also.

Figure 3.13f plots the predictive power of the filters of the ePPR model without time interaction (Figure 3.13a), the filters of the ePPR model with time interaction (Figure 3.13c), and the MID filters (Figure 3.13f), using a second-order multi-dimensional polynomial (Section 3.7.12) as the predictive model. The predictions of the filters of ePPR model without time interaction are significantly better than those of the ePPR model with time interaction (Wilcoxon signed-rank test, $p < 0.01$, Section 3.7.13), and significantly better than those of the MID filters ($p < 0.01$).

We see that for temporal correlated stimuli, which reduce the number of effective stimulus, ePPR models without time interaction perform better than ePPR models with time interaction.

3.8.7 Population plot

Figure 3.14a plots the correlation coefficients between mean cell responses to natural stimuli and predictions of ePPR models without time interaction, as a function of the maximal correlation coefficient between pairs of responses to repetitions of the stimuli⁸. Only cells for which the maximal correlation coefficient between pairs of responses was greater than 0.1 are shown. From these cells, the cortical complex cell studied in this article is the one for which we obtained best correlation coefficients (filled black circle), but similar correlation coefficients, and qualitatively similar ePPR estimates, were obtained for other complex cells. Figures 18b and 18c show the ePPR filters and non-linear functions, respectively, of the cell with the lowest correlation coefficient between cell responses and model prediction, illustrating that, even for complex cells with very noisy responses, ePPR recovered well-structured models with parameters consistent with previous findings.

The maximal correlation coefficient between pair of responses to repetitions of the stimuli is inversely proportional to the noise level in the cell responses. As expected,

⁸The simple cell plotted as having a maximal correlation between repetition of zero was probed with only one repetition of the stimuli, so we could not compute the maximal correlation between repetitions.

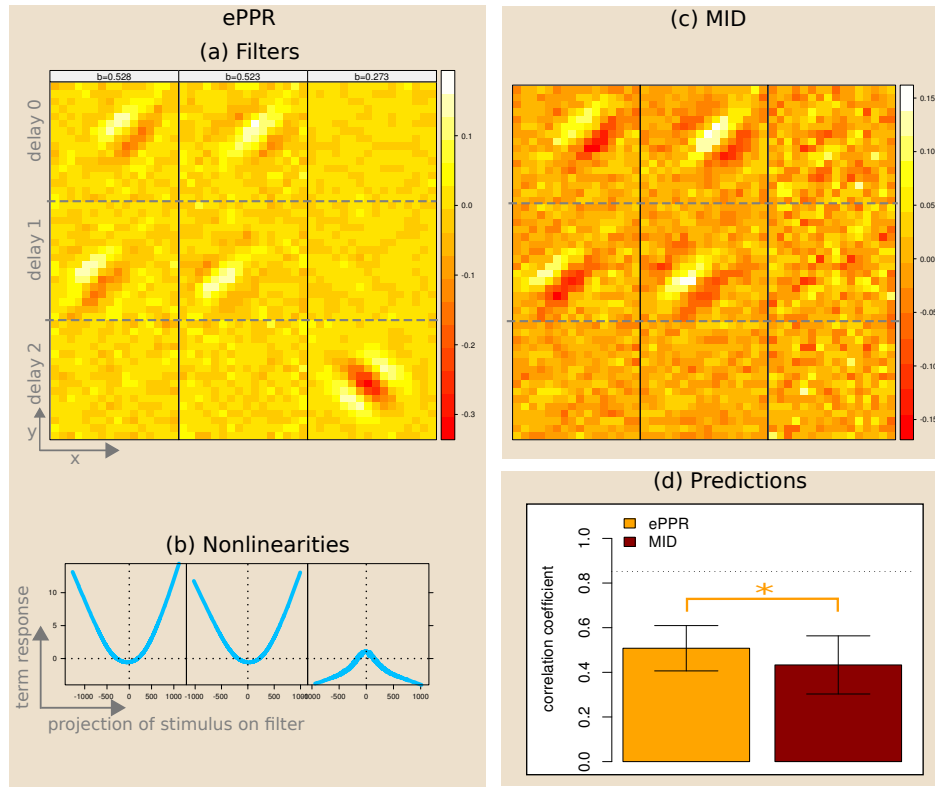


Figure 3.13: Temporally correlated inputs. (a,b): filters (a) and nonlinear functions (b) of ePPR model without time interactions. (c,d): filter (c) and nonlinear function (d) of the ePPR model with time interactions. (e): MID filters. (f): predictive power of the different filters. Light red line with asterisks indicate that the model on the left generated better predictions than the model on the right. High correlations in the inputs substantially degrade the quality of the ePPR and MID estimates. Due to the reduced number of effective stimuli, ePPR models without time interaction yielded better estimates than ePPR models with time interaction. Also, predictions from filters of ePPR models without timer interactions were significantly better than those from filters of ePPR models with time interaction, and those from MID filters.

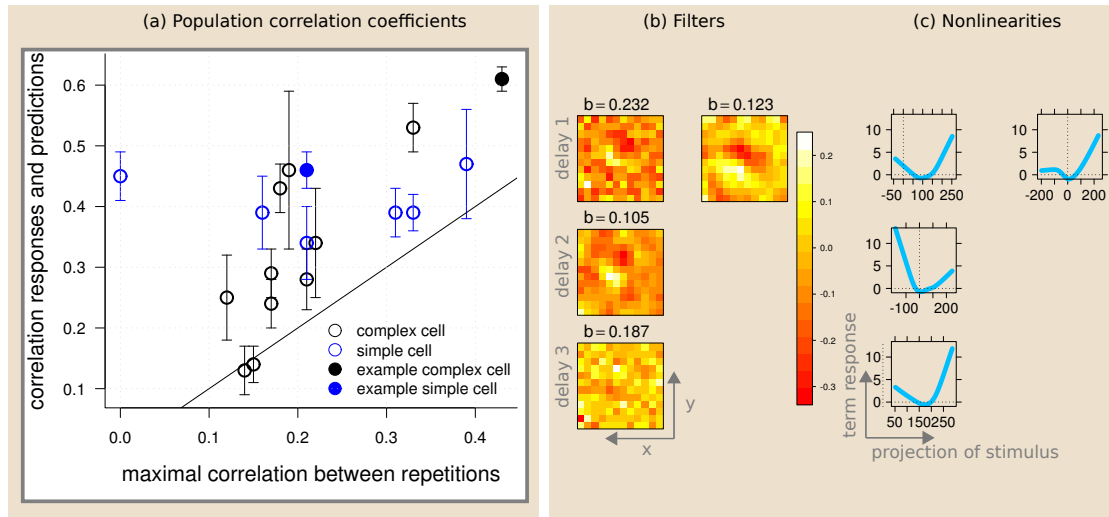


Figure 3.14: Population results: (a): correlation coefficients between mean cell responses to natural stimuli and predictions from ePPR models without time interaction, as a function of the maximal correlation coefficient between pairs of responses to repetitions of the stimuli. (b,c): filters (b) and nonlinear functions (c) of the ePPR model without time interaction estimated from responses of the cell with the lowest correlation coefficient between cell responses and models predictions. Black circles: complex cells. Blue circles: simple cells. Solid circles: example cells shown in this article. Only cells for which the maximal correlation coefficient between pairs of responses was greater than 0.1 are shown. The cortical complex cell studied in this article is the one for which we obtained best correlation coefficients, but similar correlation coefficients, and qualitatively similar ePPR estimates, were obtained for other complex cells. For example, the filters and nonlinear functions of the cell with the lowest correlation coefficient between mean cell responses and predictions from ePPR models are qualitatively similar to those of the complex cell studied in this article (Figures 15a and 15b).

Figure 3.14 shows that the predictive power of ePPR models increases as the noise level in the cell responses decreases.

The solid line shows the points where the correlation between cell responses and model predictions equal the maximal correlation between repetitions. Because we used the mean response to several repetitions of the stimuli as the cell response, the correlation coefficient between this mean cell response and the ePPR model prediction can be, and in almost all cases is, larger than the maximal correlation between individual repetitions.

Chapter 4

DISCUSSION

In this thesis we developed two methods for the characterization of visual cells from their responses to natural stimuli. In Chapter 2 we described the Volterra Relevant Space Technique (VRST) that allows the estimation of spatial Volterra models from responses of visual cells to arbitrary, including natural, stimuli. Volterra models have too many parameters to directly estimate them from limited amounts of physiological data. To overcome this limitation the VRST uses a substantiated hypothesis stating that the response of visual cells depends on a specially low-dimensional subspace of the image space. To estimate spatial low-dimensional subspaces we used the Projection Pursuit Regression (PPR) algorithm. We showed that spatial low-dimensional subspaces estimated by PPR and spatial Volterra models estimated by the VRST compared favorably with low-dimensional subspaces and with nonlinear models estimated by state-of-the-art methods. Neglecting temporal properties of the response generation mechanism is a good first approximation, but in most settings responses of visual cells are not spatial, but spatio-temporal. So, in Chapter 3 we introduced the Extended Projection Pursuit Regression (ePPR) algorithm. ePPR extends PPR to allow the estimation of spatio-temporal low-dimensional subspaces, and overcomes limitations of PPR with natural stimuli. With simulated and physiological data we demonstrated the feasibility of ePPR and we showed that it compares favorably with current methods for the estimation of low-dimensional subspaces. In addition, ePPR models estimated from responses of cortical cells to matched

natural and artificial stimuli displayed important differences, showing that the observable response properties of visual cells depend on the statistical properties of the stimuli used to probe them. Thus, using natural stimuli for the characterization of visual cells is relevant to understand natural vision.

Scientists that characterize response properties of visual cells from stimuli/response data are currently divided into two groups: those that advocate the use of artificial stimuli (refer to Rust & Movshon, 2005, for a discussion of this point of view), and those supporting the use of natural stimuli (refer to Felsen & Dan, 2005, for a discussion of this point of view). The main argument of the artificial stimuli group against the use of natural stimuli is that *“this use of natural images is so fraught with difficulty that it is not useful”* (Rust & Movshon, 2005). In this thesis we showed that the use of natural stimuli to characterize visual cells is not only feasible (models of cortical cells estimated using natural stimuli were very consistent with previous characterizations using artificial stimuli) but essential if we want to understand how visual cells operate in their natural operating conditions (models estimated using natural stimuli showed important differences with respect to models estimated using artificial stimuli).

Recent increases in the amount of digitally collected data –from the Internet, to sensor networks, to physiology– has lead to the development of novel statistical algorithms that can well approximate a very large class of functions. One wonders how much will these algorithm be able to learn about sensory cells, and what will be the role of human intelligent input in this learning. We believe that these algorithms will be able to learn a lot about sensory cells, and that they will reveal aspects of sensory processing that have remained occluded by the use of simple parametric models estimated using artificial stimuli. However, in Chapter 1 we described the bias/variance dilemma for the estimation of models from input/output data. This is the dilemma between large bias in the estimation of parametric models, and large variance in the estimation of non-parametric models. We mentioned that a solution to this dilemma is the introduction of properly designed biases into non-parametric models. In our opinion, it will be extremely difficult, if

not impossible, for machines to autonomously learn these proper biases, for which human intelligent input is essential. Moreover, we are skeptical that these algorithms will ever be smart enough to discover such unique models as the one of the double helix structure of the DNA molecule (Watson & Crick, 1953), or the one describing the initiation and propagation of action potentials in neurons (Hodgkin & Huxley, 1952).

Below we outline a few directions of future research motivated by this thesis.

Population analysis of difference between natural and random stimulation

In Chapter 3 we showed that a simple and a complex cell in primary visual cortex displayed important differences in their response properties when they were probed with natural or with random stimuli. On a second paper we will describe these differences across the population of cells recorded by Felsen et al. (2005). A complication for this study is that most complex cells do not respond well to the random stimuli used to stimulate them. Then, the comparison between a model well fitted to natural responses and a model poorly fitted to random data is not trivial.

Model selection for ePPR In Chapter 3 we mentioned that, to remove spurious terms, ePPR uses a model selection procedure based on cross-validation. For the large amounts of data recorded from cortical cells this procedure worked reliably. However, large data sets are not very common in sensory neurophysiology. So, it is important to evaluate model selection procedures that do not require cross-validated data and work with smaller datasets, e.g., Akaike information criterion (Akaike, 1974) or Bayesian information criterion (Schwarz, 1978). The estimation of the filters and that of the nonlinear functions of ePPR models are regularized, so they can be performed with smaller datasets. The model selection procedure is the only component of the ePPR algorithm that requires large data sets. Thus, with a model selection procedure that does not require cross-validated data, we will be able to characterize many more visual cells for which only smaller datasets are available.

Spatio-temporal Volterra models As in the spatial case, one could use the spatio-temporal relevant dimensions obtained by ePPR to estimate spatio-temporal Volterra models. Spatio-temporal Volterra models could reveal nonlinear interactions at the pixel level that are not evident in ePPR models. However, Volterra models estimated through the VRST can have large variability, and one should develop statistical methods to assess this variability.

Assessing the variability in estimated Volterra and ePPR models To check the variability in ePPR estimates, for each cell we estimated five ePPR models, from different resampled subsets of the training data. We did not observe large variability in these estimates. However, a more rigorous statistical analysis of variability, for example using bootstrap techniques (Efron & Tibshirani, 1993; Shao & Tu, 1995), is pertinent. A complication is that most bootstrap techniques require independent and identically distributed data and the responses of cells analyzed in this thesis are highly correlated, and therefore non-independent.

Non-parametric models for populations of cells A natural extension of the research in this thesis is to move from the characterization of single cells to the characterization of a simultaneously recorded population of cells. Recently, Pillow et al. (2008) made a first contribution along this direction, by using a semi-parametric model to characterize a population of retinal ganglion cells from their responses to artificial stimuli. It would be useful to extend this work by estimating a more general non-parametric model, and by using natural stimuli. The PPR algorithm, as described in Friedman (1984a), can be used to estimate a model of the responses of multiple cells. Although this model may be too restrictive, it is a good starting point for a more general non-parametric model of populations of visual cells that can be estimated using natural stimuli.

Application of the VRST and ePPR to other cortical systems In this thesis we used the VRST and the ePPR algorithm to characterize cells in the visual system.

But, these methods are general and can be applied to characterize cells in other sensory systems. Moreover, Volterra models are currently being used to characterize hippocampal cells (Song, Marmarelis, & Berger, 2009; Song, Wang, Marmarelis, & Berger, 2009). These Volterra models are estimated using a low-dimensional subspace, as in the VRST. But this low-dimensional subspace is constructed using parametric Laguerre basis functions, instead of being learned non-parametrically from the data, as in ePPR. Then, it would be useful to check how Volterra models estimated using Laguerre basis functions compare with those estimated using basis functions learned non-parametrically from the data, for the characterization of cells in the hippocampus.

The research that lead to this thesis has been delightful. It combined beautiful data, with powerful algorithms, and with elegant mathematical proofs. It is astonishing to hear the preferences of something so minuscule as a visual cell responding to stimuli that it likes or dislikes. Powerful algorithms applied to physiological data provide new glasses to observe previously hidden aspects of nature. And the perfection of a mathematical proof demonstrating a property of an algorithm that one has developed is dazzling.

References

- Adelson, E., & Bergen, J. (1985). Spatiotemporal energy models for the perception of motion. *Journal of the Optical Society of America A*, 2, 284-299.
- Aggelopoulos, N., Franco, L., & Rolls, E. (2005). Object perception in natural scenes: encoding by inferior temporal cortex simultaneously recorded neurons. *Journal of Neurophysiology*, 93(3), 1342-1357.
- Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6), 716-723.
- Albrecht, D., & Hamilton, D. (1982). Striate cortex of monkey and cat: contrast response function. *Journal of Neurophysiology*, 48, 217-237.
- Albrecht, D., & Valois, R. D. (1981). Striate cortex responses to periodic patterns with and without the fundamental harmonics. *Journal of Physiology of London*, 319, 497-514.
- Amorocho, J., & Brandstetter, A. (1971). Determination of nonlinear functional response functions in rainfall-runoff processes. *Water Resources Research*, 7, 1087-1101.
- Amthor, F., Grzywacz, N., & Merwine, D. (1996). Extra-receptive-field motion facilitation in on-off directionally selective ganglion cells of the rabbit retina. *Visual Neuroscience*, 13(2), 303-309.
- Baccus, S., & Meister, M. (2002). Fast and slow contrast adaptation in retinal circuitry. *Neuron*, 36(5), 909-919.
- Balboa, R., & Grzywacz, N. (2000). The role of early lateral inhibition: more than maximizing luminance information. *Vision Research*, 17, 77-89.
- Barlow, H. (1953). Summation and inhibition in the frog's retina. *Journal of Physiology (London)*, 119, 69-88.
- Barlow, H. (1961). Possible principles underlying the transformations of sensory messages. In *Sensory communication* (p. 217-234). MIT Press.

- Bellman, R. (1961). *Adaptive control processes*. Princeton, NJ: Princeton University Press.
- Ben-Israel, A., & Greville, T. (1980). *Generalized inverses: Theory and applications*. Huntington, NY: Krieger.
- Bertsekas, D. (1999). *Nonlinear programming* (2nd ed.). Nashua, NH: Athena Scientific.
- Bialek, W., & de Ruyter van Steveninck, R. (2005). *Features and dimensions: Motion estimation in fly vision*. <http://arxiv.org/abs/q-bio/0505003>.
- Bonds, A. (1989). Role of inhibition in the specification of orientation selectivity of cells in the cat striate cortex. *Visual Neuroscience*, *2*, 41-55.
- Bose, A. (1956). *A theory of nonlinear systems* (Tech. Rep. No. 309). Cambridge, MA: Research Laboratory of Electronics, M.I.T.
- Brenner, N., Bialek, W., & de Ruyter van Steveninck, R. (2000). Adaptive rescaling optimizes information transmission. *Neuron*, *26*, 695-702.
- Brown, J., & Churchill, R. (1996). *Complex variables and applications* (Sixth ed.). New York, NY: McGraw-Hill, Inc.
- Burnham, K., & Anderson, D. (2002). *Model selection and multimodel inference: a practical information-theoretic approach*. Berlin, Germany: Springer.
- Carandini, M., & Heeger, D. (1994). Summation and division by neurons in primate visual cortex. *Science*, *264*, 1333-1336.
- Chen, X., Han, F., Poo, M., & Dan, Y. (2007). Excitatory and suppressive receptive field subunits in awake monkey primary visual cortex (V1). *PNAS*, *104*(48), 19120-19125.
- Chichilnisky, E. (2001). A simple white noise analysis of neuronal light response. *Network: Computation in Neural Systems*, *12*, 199-213.
- Christianson, G., Sahani, M., & Linden, J. (2008). The consequences of response nonlinearities for interpretation of spectrotemporal receptive fields. *The Journal of Neurosciences*, *28*(2), 446-455.
- Citron, M., Kroeker, J., & McCann, G. (1981). Nonlinear interactions in ganglion cell receptive fields. *Journal of neurophysiology*, *46*, 1161-1176.
- Dan, Y., Attick, J., & Reid, R. (1996). Efficient coding of natural scenes in the lateral geniculate nucleus: experimental test of a computational theory. *The Journal of*

Neuroscience, 16, 3351-3362.

- David, S., Mesgarani, N., Fritz, J., & Shamma, S. (2009). Rapid synaptic depression explains nonlinear modulation of spectro-temporal tuning in primary auditory cortex by natural stimuli. *The Journal of Neuroscience*, 29(11), 3374-3386.
- David, S., Vinje, W., & Gallant, J. (2004). Natural stimulus statistics alter the receptive field structure of V1 neurons. *The Journal of Neuroscience*, 24(31), 6991-7006.
- de Boer, E., & Kuyper, P. (1968). Triggered correlation. *IEEE Transactions on Biomedical Engineering*, 15, 169-179.
- de Ruyter van Steveninck, R., & Bialek, W. (1988). Real-time performance of a movement-sensitive neuron in the blowfly visual system: coding and information transfer in short spike sequences. *Proceedings Royal Society of London B*, 234, 379-414.
- Dean, A. (1981). The relationship between response amplitude and contrast for cat striate cortical neurones. *Journal of Physiology (London)*, 318, 1255-1267.
- Dean, A., & Tolhurst, D. (1986). Factors influencing the temporal phase of response to bar and grating stimuli for simple cells in the cat striate cortex. *Experimental Brain Research*, 62, 143-151.
- DeAngelis, G., Ohzawa, I., & Freeman, R. (1993a). Spatiotemporal organization of simple-cell receptive fields in the cat's striate cortex II. Linearity of temporal and spatial summation. *Journal of Neuroscience*, 69, 1118-1135.
- DeAngelis, G., Ohzawa, I., & Freeman, R. (1993b). Spatiotemporal organization of simple-cell receptive fields in the cat's striate cortex I. General characteristics and postnatal development. *Journal of Neuroscience*, 69, 1091-1117.
- Diaconis, P., & Shahshahani, M. (1984). On nonlinear functions of linear combinations. *SIAM Journal on Scientific Computing*, 5(1), 175-191.
- Donoho, D., Johnstone, I., Rousseeuw, P., & Stahel, W. (1985). Discussion: Projection pursuit. *The Annals of Statistics*, 13(2), 496-500.
- Efron, B., & Tibshirani, R. (1993). *An introduction to the bootstrap*. New York, NY: Chapman & Hall.
- Emerson, R., Citron, M., Vaughn, W., & Klein, S. (1987). Nonlinear directionally selective subunits in complex cells of a cat striate cortex. *Journal of Neurophysiology*, 58, 33-65.

- Felsen, G., & Dan, Y. (2005). A natural approach to studying vision. *Nature neuroscience*, *8*(12), 1643-1646.
- Felsen, G., Touryan, J., Han, F., & Dan, Y. (2005, 9). Cortical sensitivity to visual features in natural scenes. *PLoS Biology*, *3*(10), e342. Available from <http://dx.doi.org/10.1371/journal.pbio.0030342>
- Field, D. (1987). Relations between the statistics of natural images and the response properties of cortical cells. *Journal of the Optical Society of America A*, *4*(12), 2379-2394.
- Friedman, J. (1984a). *Smart user's guide* (Tech. Rep. No. 1). Palo Alto, CA: Department of Statistics, Stanford University.
- Friedman, J. (1984b). *A variable span scatterplot smoother* (Tech. Rep. No. 5). Palo Alto, CA: Department of Statistics, Stanford University.
- Friedman, J., & Stuetzle, W. (1981). Projection pursuit regression. *Journal of the American Statistical Association*, *76*(376), 817-823.
- Geladi, P., & Kowalski, B. (1986). Partial least-squares regression: a tutorial. *Analytica Chimica Acta*, *185*, 1-17.
- Geman, S., Bienenstock, E., & Doursat, R. (1992). Neural networks and the bias/variance dilemma. *Neural Computation*, *4*, 1-58.
- Golub, G., & van Loan, C. (1996). *Matrix computations*. Baltimore, MD: The Johns Hopkins University Press.
- Green, P., & Silverman, B. (1994). *Nonparametric regression and generalized linear models*. Boca Raton, FL: CRC Press.
- Grzywacz, N., & Balboa, R. (2002). A bayesian framework for sensory adaptation. *Neural Computation*, *14*, 543-559.
- Guo, K., Robertson, R., Mahmoodi, S., & Young, M. (2005). Center-surround interactions in response to natural scene stimulation in the primary visual cortex. *European Journal of Neuroscience*, *21*(2), 536-548.
- Hansen, P. (1987). The truncated SVD as a method for regularization. *BIT*, *27*(4), 534-553.
- Hartline, H. (1940). The receptive fields of optic nerve fibers. *American physiological society*, *130*(4), 690-9.

- Heeger, D. (1992). Normalization of cell responses in cat striate cortex. *Visual Neuroscience*, 9, 181-97.
- Helland, I. (1988). On the structure of partial least squares regression. *Communications in Statistics: Simulation and Computation*, 17(2), 584-607.
- Hertz, J., Palmer, R., & Krogh, A. (1991). *Introduction to the theory of neural computation*. Redwood City, CA: Addison-Wesley.
- Hodgkin, A., & Huxley, A. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *Journal of Physiology*, 117, 500-544.
- Hollander, M., & Wolfe, D. (1999). *Nonparametric statistical methods* (Second ed.). New York, NY: Wiley-Interscience.
- Hubel, D., & Wiesel, T. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *Journal of Physiology*, 160, 106-154.
- Huber, P. (1985). Projection pursuit. *The Annals of Statistics*, 13(2), 435-475.
- Hwang, J., Lay, S., Maechler, M., Martin, R., & Schimert, J. (1994). Regression modelling in back-propagation and projection pursuit learning. *IEEE Transactions on Neural Networks*, 5(3), 342-353.
- Jones, J., Stepnoski, A., & Palmer, L. (1987). The two-dimensional spectral structure of simple receptive-fields in cat striate cortex. *Journal of Neurophysiology*, 58(6), 1212-1232.
- Jones, L. (1987). On a conjecture of Huber concerning the convergence of projection pursuit regression. *The Annals of Statistics*, 15(2), 880-882.
- Kaplan, E., & Benardete, E. (2001). The dynamics of primate retinal ganglion cells. *Progress in brain research*, 134, 17-34.
- Korenberg, M., & Hunter, I. (1986). The identification of nonlinear biological systems: LNL cascade models. *Biological Cybernetics*, 55, 125-134.
- Kuffler, S. (1953). Discharge patterns and functional organization of mammalian retina. *Journal of Neurophysiology*, 16, 37-68.
- Lee, Y. (1964). Contributions of norbert wiener to linear theory and nonlinear theory in engineering. In *Selected papers of norbert wiener* (p. 17-33). Cambridge, MA: MIT Press.

- Lee, Y., & Schetzen, M. (1965). Measurement of the wiener kernels of a nonlinear system by cross-correlation. *International Journal of Control*, *2*, 237-254.
- Li, K. (1991). Sliced inverse regression for dimension reduction (with discussion). *Journal of the American Statistical Association*, *86*, 316-342.
- Liberman, M. (1982). The cochlear frequency map for the cat: labeling auditory-nerve fibers of known characteristic frequency. *Journal of the acustical society of America*, *72*, 1441-9.
- Livingstone, M., & Conway, B. R. (2003). Substructure of direction-selective receptive fields in macaque V1. *Journal of Neurophysiology*, *89*, 2743-2759.
- Maffei, L., & Fiorentini, A. (1973). The visual cortex as a spatial frequency analyzer. *Vision Research*, *13*, 1255-1267.
- Mante, V., Bonin, V., & Carandini, M. (2008). Functional mechanisms shaping lateral geniculate responses to artificial and natural stimuli. *Neuron*, *58*(4), 625-638.
- Maravall, M., Petersen, R., Fairhall, A., Arabzadeh, E., & Diamond, M. (2007). Shifts in coding properties and maintenance of information transmission during adaptation in barrel cortex. *PLoS Biology*, *5*, e19.
- Marmarelis, P., & Marmarelis, V. (1978). *Analysis of physiological systems: the white-noise approach*. New York, NY: Plenum Press.
- Marmarelis, P., & Naka, K. (1972). White-noise analysis of a neuron chain: An application of the wiener theory. *Science*, *175*, 1276-1278.
- Marmarelis, V. (1993). Identification of nonlinear biological systems using Laguerre expansions of kernels. *Annals of Biomedical Engineering*, *21*, 573-589.
- Marmarelis, V. Z. (2004). *Nonlinear dynamic modeling of physiological systems*. Hoboken, NJ: Wiley.
- Mechler, F., & Ringach, D. (2002). On the classification of simple and complex cells. *Vision Research*, *42*(8), 1017-33.
- Mendel, J. (1995). *Lessons in estimation theory for signal processing, communications, and control*. Upper Saddle River, NJ: Prentice Hall.
- Mitsis, G., & Marmarelis, V. (2002). Modeling of nonlinear physiological systems with fast and slow dynamics. i. methodology. *Annals of Biomedical Engineering*, *30*, 272-281.

- Morrone, M., Burr, D., & Maffei, L. (1982). Functional implications of cross-orientation inhibition of cortical visual cells. 1. neurophysiological evidence. *Proceedings of the Royal Society of London B.*, *216*, 335-354.
- Movshon, J., Thompson, I., & Tolhurst, D. (1978a). Receptive field organization of complex cells in the cat's striate cortex. *Journal of Physiology*, *383*, 79-99.
- Movshon, J., Thompson, I., & Tolhurst, D. (1978b). Spatial summation in the receptive fields of simple cells in the cat's striate cortex. *Journal of Physiology*, *383*, 53-77.
- Nocedal, J., & Wright, S. (2006). *Numerical optimization*. New York, NY: Springer.
- Nykamp, D., & Ringach, D. (2002). Full identification of a linear-nonlinear system via cross-correlation analysis. *Journal of Vision*, *2*(1), 1-11.
- Palm, G., & Poggio, T. (1977). The volterra representation and the wiener expansion: validity and pitfalls. *SIAM Journal on Applied Mathematics*, *33*(2), 195-216.
- Paninski, L. (2003). Convergence properties of three spike-triggered analysis techniques. *Network: Computation in Neural Systems*, *14*, 437-464.
- Pillow, J., Shlens, J., Paninski, L., Sher, A., Litke, A., Chichilnisky, E., et al. (2008). Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, *454*(7207), 995-999.
- Prenger, R., Wu, M.-K., David, S., & Gallant, J. (2004). Nonlinear V1 responses to natural scenes revealed by neural network analysis. *Neural Networks*, *17*, 663-679.
- Rapela, J., Felsen, G., Touryan, J., Mendel, J., & Grzywacz, N. (2010). ePPR: a new strategy for the characterization of sensory cells from input/output data. *Network: Computation in Neural Systems*. (In press.)
- Rapela, J., Mendel, J., & Grzywacz, N. (2006). Estimating nonlinear receptive fields from natural images. *Journal of Vision*, *6*(4), 441-474.
- Ringach, D., Hawken, M., & Shapley, R. (2002). Receptive field structure of neurons in monkey primary visual cortex revealed by stimulation with natural image sequences. *Journal of Vision*, *2*(1), 12-24.
- Ringach, D., Sapiro, G., & Shapely, R. (1997). A subspace reverse -correlation technique for the study of visual neurons. *Vision Research*, *37*, 2455-2464.
- Roosen, C. (1994). *Implementation of Automatic Smoothing Spline Projection Pursuit (Roosen & Hastie, 1994)*. (<http://lib.stat.cmu.edu/jcgs/roosen-hastie>)

- Roosen, C., & Hastie, T. (1994). Automatic smoothing spline projection pursuit. *Journal of Computational and Graphical Statistics*, 3(3), 235-248.
- Ruderman, D. L., & Bialek, W. (1994). Statistics of natural images: scaling in the woods. *Physical Review Letters*, 76(6), 814-817.
- Rudin, W. (1976). *Principles of mathematical analysis* (3rd ed.). Columbus, OH: McGraw-Hill.
- Rust, N., & Movshon, J. (2005). In praise of artifice. *Nature neuroscience*, 8(12), 1647-1650.
- Rust, N., Schwartz, O., Movshon, J., & Simoncelli, E. (2005). Spatiotemporal elements of macaque V1 receptive fields. *Neuron*, 46, 945-956.
- Scharf, L. (1991). The SVD and reduced-rank signal processing. In R. Vaccaro (Ed.), *SVD and signal processing, II: algorithms, analysis and applications* (p. 3-31). New York, NY: Elsevier.
- Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics*, 6(2), 461-464.
- Scialoja, G., & Freeman, R. (1982). Orientation selectivity in the cat's striate cortex is invariant with stimulus contrast. *Experimental Brain Research*, 46(3), 457-461.
- Shao, J., & Tu, D. (1995). *The jackknife and bootstrap*. New York, NY: Springer.
- Sharpee, T. (2007). Comparison of information and variance maximization strategies for characterizing neural feature selectivity. *Statistics in medicine*, 26, 4009-4031.
- Sharpee, T., Miller, K., & Stryker, M. (2008). On the importance of static nonlinearity in estimating spatiotemporal neural filters with natural stimuli. *Journal of Neurophysiology*, 99, 2496-2509.
- Sharpee, T., Rust, N., & Bialek, W. (2004). Analyzing neural responses to natural signals: maximally informative dimensions. *Neural computation*, 16, 223-250. (Code available at <http://cnl-t.salk.edu/Download/>.)
- Sharpee, T., Sugihara, H., Kurgansky, A., Rebrik, S., Stryker, M., & Miller, K. (2006). Adaptive filtering enhances information transmission in visual cortex. *Nature*, 439(7079), 936-942.
- Simoncelli, E., & Olshausen, B. (2001). Natural image statistics and neural representations. *Annual Review of Neuroscience*, 24, 1193-216.

- Simoncelli, E., Paninski, L., Pillow, J., & Schwartz, O. (2004). Characterization of neural responses with stochastic stimuli. In M. Gazzaniga (Ed.), *The new cognitive neurosciences* (3rd ed., p. 327-338). Cambridge, MA: MIT Press.
- Smyth, D., Willmore, B., Baker, G., Thompson, I., & Tolhurst, D. (2003). The receptive-field organization of simple cells in primary visual cortex of ferrets under natural scene stimulation. *The Journal of Neuroscience*, *23*, 4746-4759.
- Song, D., Marmarelis, V., & Berger, T. (2009). Parametric and non-parametric modeling of short-term synaptic plasticity. part i: computational study. *Journal of Computational Neuroscience*, *26*(1), 1-19.
- Song, D., Wang, Z., Marmarelis, V., & Berger, T. (2009). Parametric and non-parametric modeling of short-term synaptic plasticity. part ii: Experimental study. *Journal of Computational Neuroscience*, *26*(1), 21-37.
- Spekreijse, H. (1969). Rectification in the goldfish retina: analysis by sinusoidal and auxiliary stimulation. *Vision Research*, *9*, 1461-1472.
- Spekreijse, H., & Oosting, J. (1970). Linearizing: a method for analyzing and synthesizing nonlinear systems. *Kybernetik*, *7*, 23-31.
- Spekreijse, H., & Reits, D. (1982). Sequential analysis of the visual evoked potential system in man: nonlinear analysis of a sandwich system. *Annals of the new York Academy of Science*, *388*, 72-97.
- Srinivasan, M., Laughlin, S., & Dubs, A. (1982). Predictive coding: a fresh view of inhibition in the retina. *Proceedings of the Royal Society of London B*, *216*, 427-459.
- Sturges, H. (1926). The choice of a class-interval. *Journal of the American Statistical Association*, *21*, 65-66.
- Tadmor, Y., & Tolhurst, D. (2000). Calculating the contrasts that retinal ganglion cells and lgn neurones encounter in natural scenes. *Vision Research*, *40*(22), 3145-3147.
- Tauchi, M., & Masland, R. (1984). The shape and arrangement of the cholinergic neurons in the rabbit retina. *Proceedings of the Royal Society of London B*, *223*(1230), 101-119.
- Theunissen, F., David, S., Singh, N., Hsu, A., Vinje, W., & Gallant, J. (2001). Estimating spatio-temporal receptive fields of auditory and visual neurons from their responses to natural stimuli. *Network: Computation in Neural Systems*, *12*, 289-316.

- Theunissen, F., Sen, K., & Doupe, A. (2000). Spectral-temporal receptive fields of nonlinear auditory neurons obtained using natural sounds. *Journal of Neuroscience*, *20*(6), 2315-31.
- Theunissen, F., & Shaevitz, S. (2006). Auditory processing of vocal sounds in birds. *Current opinion in neurobiology*, *16*, 400-407.
- Touryan, J., Felsen, G., & Dan, Y. (2005). Spatial structure of complex cell receptive fields measured with natural images. *Neuron*, *45*, 781-791.
- Touryan, J., Lau, B., & Dan, Y. (2002). Isolation of relevant visual features from random stimuli for cortical complex cells. *Journal of Neuroscience*, *22*(24), 10811-10818.
- van Hateren, J., & Ruderman, D. (1998). Independent component analysis of natural image sequences yields spatio-temporal filters similar to simple cells in primary visual cortex. *Proceedings Royal Society London B*, *265*, 2315-2320.
- van Hateren, J., & van der Schaaf, A. (1998). Independent component filters of natural images compared with simple cells in primary visual cortex. *Proceedings Royal Society London B*, *265*, 359-366. (<http://hlab.phys.rug.nl/imlib/index.html>)
- Venables, W., & Smith, D. (2002). *An introduction to R*. Network Theory Ltd. (<http://www.r-project.org/>)
- Victor, J., & Shapley, R. (1980). A method of nonlinear analysis in the frequency domain. *Biphasical Journal*, *29*, 459-484.
- Vinje, W., & Gallant, J. (2000). Sparse coding and decorrelation in primary visual cortex during natural vision. *Science*, *287*(5456), 1273-1276.
- Vinje, W., & Gallant, J. (2002). Natural stimulation of the nonclassical receptive field increases information transmission in V1. *Journal of Neuroscience*, *22*(7), 2904-2915.
- Volterra, V. (1930). *Theory of functionals and of integral and integro-differential equations*. New York: Dover Publications.
- Wang, X., Wei, Y., Vaingankar, V., Wang, Q., Koepsell, K., Sommer, F., et al. (2007). Feedforward excitation and inhibition evoke dual modes of firing in the cat's visual thalamus during naturalistic viewing. *Neuron*, *55*(3), 465-478.
- Watanabe, A., & Stark, L. (1975). Kernel method for nonlinear analysis: identification of biological control system. *Mathematical Biosciences*, *27*, 99-108.

- Watson, J., & Crick, F. (1953). Molecular structure of nucleic acids: a structure for deoxyribose nucleic acid. *Nature*, *171*(4356), 737-738.
- Weiss, T. (1966). A model of the peripheral auditory system. *Kybernetik*, *3*, 153-175.
- Weliky, M., Fiser, J., Hunt, R., & Wagner, D. (2003). Coding of natural scenes in primary visual cortex. *Neuron*, *37*(4), 703-718.
- Wiener, N. (1958). *Nonlinear problems in random theory*. New York: Wiley.
- Willmore, B., & Smyth, D. (2003). Methods for first-order kernel estimation: simple-cell receptive fields from responses to natural scenes. *Network: computation in neural systems*, *14*, 553-577.
- Wooley, S., Gill, P., & Theunissen, F. (2006). Stimulus-dependent auditory tuning results in synchronous population coding of vocalization in the songbird midbrain. *Journal of Neuroscience*, *26*(9), 2499-512.
- Woolley, S., Fremouw, T., Hsu, A., & Theunissen, F. (2005). Tuning for spectro-temporal modulations as a mechanism for auditory discrimination of natural sounds. *Nature Neuroscience*, *8*, 1371-1379.
- Wu, M., David, S., & Gallant, J. (2006). Complete functional characterization of sensory neurons by system identification. *Annual Review of Neuroscience*, *29*, 477-505.
- Zhu, S., & Mumford, D. (1997). Prior learning and gibbs reaction-diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *19*(11), 1236-1250.