Visual Temporal Prediction: Representation, Estimation, and Modeling

by

Pierre-Étienne H. Fiquet

A dissertation submitted in partial fulfillment
of the requirements for the degree of

Doctor of Philosophy

Center for Neural Science
New York University

September, 2024

Eero P. Simoncelli

Acknowledgements

I am deeply grateful to my advisor, Eero Simoncelli, for his generous and steady support throughout my journey in graduate school. His profound and creative thinking has been a great source of inspiration and delight. I admire the scientist for his ability to reason and see connections in a wide landscape of ideas, to distill those findings into simple insights that make a real difference, and to communicate those questions and insights with clarity and elegance. I am indebted to the caring mentor who shared the joy of doing research with me, and who guided me while also giving me the freedom to explore. In return, I aspire to transmit further this enthusiasm for finding things out.

I was lucky to receive advice from a number of exceptional scholars. Thanks to Wei Ji Ma, Tony Movshon, Mitya Chklovskii, and Ann Hermundstad for serving on my thesis committee and for their feedback that helped improve the work presented here. Thanks to my fantastic teachers at NYU and Columbia: Afonso Bandeira, Tim Austin, Joan Bruna, Carlos Fernandez-Granda, and David Blei.

The Laboratory for Computational Vision is an outstanding environment, conducive to asking ambitious questions, and I was fortunate to spend time with many brilliant and kind lab-mates. Thanks to Alex, Olivier, and Jimmy for indulging my curiosity early on. Thanks to Paul, Kede, Tim, Billy, Kate, Sreyas, Colin, Jing, Hope, Julie, and Ben for contributing to this friendly and collaborative environment. Special thanks to my office neighbors and friends, Caroline, Zahra, Nikhil, Lyndon, Teddy, Xinyuan, Jenelle, David, and Florentin, for

our many conversations over the blackboards. I also appreciate the staff at NYU, Amala, Jess, Jose, and Asa; and the staff at the Flatiron Institute, Matthew, Jess, and Noah, for their work and indispensable help.

Many scientists outside of LCV had an impact on my work. Many thanks to Elad Schneidman, Stephane Mallat, and David Brainard for their visits which have been influential and stimulating to me. Thanks to my mentors at ENS in Paris and Champalimaud in Lisbon, Jean-Pierre Nadal, Emmanuel Dupoux, Amaury Lambert, Sophie Denève, Srdjan Ostojic, and Christian Machens, for introducing me to and helping me switch to the path of scientific research.

New York became a new home because of the wonderful people I met outside the lab. I am very grateful to Alaina, Arthur, Alvaro, and Jules for their love and friendship. I would like to acknowledge my family, to whom I owe everything. In particular, my mother, my father, and my stepmother have been an unwavering source of love, care, and encouragement. Finally, my gratitude goes to my partner, Machee, whose love, humor, and intelligence make building our life together a pure joy.

Abstract

All organisms make temporal predictions, and their evolutionary fitness generally depends on the accuracy of these predictions. Understanding what structure enables computing temporal predictions in complex natural scenarios is central to computational neuroscience and machine learning. This thesis focuses on visual processing and describes a unified approach for the representation and estimation of dynamic visual signals, as computed with neural elements in brains or machines. We propose that temporal prediction can serve as a general objective function, both for unsupervised learning of visual representations and for estimating probable future frames under uncertainty. Optimizing for next-frame prediction leverages the order of time, especially visual motion, and extracts predictive information from image sequences, without requiring labeled data. The architecture of a predictive system plays a critical role and we hypothesize that it should reflect the symmetry properties of the physical world. Specifically, a model that discovers the transformations acting in a visual scene should exploit these transformations to predict accurately and should remain agnostic when these transformations are ambiguous. Such interpretable models can serve as guides to explain the responses of neurons in sensory cortices and their functional role in visual perception.

We first describe the empirical structure of dynamic visual scenes and then develop a mathematical theory for exploiting that structure. The movement of observers and objects creates distinct temporal structures in visual signals, allowing for partial prediction of future signals based on past ones. Motivated by group representation theory, we propose a method

to discover and utilize the transformation structures of image sequences and show that local phase measurements play a fundamental role. The proposed model extrapolates visual signals in a local polar representation, this representation is learned via next-frame prediction. This polar prediction model successfully recovers simple transformations in synthetic datasets and scales to natural image sequences. The architecture is simple yet effective: it contains a single hidden stage with one non-linearity that factorizes slow form and steady motion signals. We demonstrate that polar prediction achieves better prediction performance than traditional approaches based on motion compensation, and that it rivals conventional deep networks trained on prediction.

We then confront the inherent uncertainty of visual temporal prediction and develop a framework for learning and sampling the conditional density of the next frame given the past few observed frames. Casting prediction as a probabilistic inference problem is motivated by the need to cope with ambiguity in natural image sequences. We describe a regression-based framework that implicitly estimates the distribution of the next frame, effectively learning a conditional image density from high-dimensional signals. This is achieved with a simple resilience-to-noise objective function: a deep neural network is trained to map to past conditioning frames and a noisy observation of the next frame to an estimated denoised next frame. The network is trained over a range of noise levels without access to that noise level, *i.e.*, it is blind and universal. This denoising objective has the desirable property of being local in the space of densities, and training across noise levels forces the network to extract information about the stable underlying distribution of probable next frame given past conditioning. We consider synthetic image sequences composed of moving disks that occlude each other and demonstrate that trained networks can handle challenging cases of bifurcating temporal trajectories—effectively choosing one occlusion or another when the observation is ambiguous. Furthermore, local linear analysis of a network trained on natural image sequences reveals that the model automatically weights evidence by reliability: the

model integrates information from past conditioning and noisy observation, adapting to the amount of predictive information in the conditioning, and the noise level in the observation.

Finally, we discuss the implications of this work for understanding biological vision. Starting from the polar prediction model, we derive a circuit algorithm composed of local neural computations for predicting natural image sequences. The circuit is composed of canonical computational elements that have received ample physiological evidence: its components resemble the normalized simple and direction-selective complex cell models of primate V1 neurons. Unlike the polar prediction model, this circuit algorithm does not impose a polar factorization, instead, it lets complex-cell-like units learn a combination of quadratic responses. Furthermore, we outline a method for gradually extracting slower and more abstract features by cascading this biologically plausible mechanism. These models offer a normative framework for understanding how the visual system represents sensory inputs in a form that simplifies temporal prediction. Together, our work on visual temporal prediction builds connections between computational modeling and brain sciences. These connections can guide the design and analysis of physiological and perceptual experiments, and can also motivate further developments in machine learning.

Contents

List of Figures

List of Appendices

Chapter 1

Introduction

> "*Se tenir sur les épaules des géants et voir plus loin. Voir dans l'invisible, à travers l'espace et à travers le temps. Plonger notre regard dans le passé et découvrir que notre passé est immense. Pouvoir remonter le temps à contre courant. Pouvoir distinguer à travers le long écoulement des âges, des éclats de passé qui soudain, resurgissent de l'oubli. Des éclats de mondes disparus. Et partir à la recherche des lointaines métamorphoses qui ont donné naissance au monde d'aujourd'hui.*"
>
> - Jean-Claude Ameisen (2012), *Sur les épaules de Darwin*

All organisms make temporal predictions, and their evolutionary fitness generally depends on the accuracy of these predictions. Simple organisms make short term predictions over low dimensional signals. For example, a bacterium following a chemical gradient measures local change, compares it to an internal state and extrapolates to decide where to go next. Or consider the rhythmic oscillations of a flie's circadian clock: it cycles through a biochemical loop and provides a signal used to anticipate physiological needs throughout the day. More complex organisms build on these strategies to predict higher dimensional signals over longer timescales while maintaining correspondingly rich internal states. For example, a teenager choosing a college major makes a consequential decision based on limited information and under considerable uncertainty. Or consider society as a whole: we adapt to our planet's changing climate and rely on probabilistic projections of future climate to inform our decisions and actions.

1.1 Temporal prediction

Temporal prediction is arguably the most important of our tasks: it is essential to our ability to survive, adapt, and reproduce. It is also a compelling computation to study: it requires combining internal state with measurements. Internally generated predictions about the external world have to be updated according to incoming evidence from sensory signals as well as information retrieved from memory.

One reason temporal prediction is important is that any sensory transduction, cognitive operation or motor action induces time lags. Computation takes time, and biophysical computation even more so. Therefore, any organism under ecological pressure to act in synchrony with or in anticipation of the other players in the environment has to predict forward in time. In that sense, organisms are not passively registering information currently available in the environment, instead they extract the parts of it that are important to guide their actions—these are the parts that might impact their future. As a result much of behavior relies on making predictions about future outcomes given recent observations. For example, physiological regulation aims to anticipate an organism's needs in order to meet them: predicting and avoiding errors, rather than correcting them after the fact [203]. Such regulation efficiently allocates energy by adapting to predicted demand, and loss of predictability is stressful. Exactitude is not the only merit of a prediction—anticipating also readies the organism to learn from events as they unfold. In fact, prediction is a form of active engagement, which, together with getting feedback (*i.e.*, prediction errors), are fundamental to any learning system [50].

A strong version of this hypothesis proposes that making prediction is the purpose of brains, and that organisms evolved more complex brains in order to perform more complex predictions. The idea is that, along evolution, simple measurement strategies and prediction primitives were reused and cascaded. Many researchers have recognized that temporal prediction has the potential to provide a unifying organizing principle for intelligence—both biological as well as artificial—and have created a vast literature across machine learning, cognitive science and neuroscience. The general idea at the center of this literature is that prediction requires abstraction: intelligence emerges in systems that learn hierarchical models of the world by predicting at increasingly long time scales and in increasingly abstract representation [128].

A fertile problem. Because it is both mathematically fascinating and practically important, the problem of temporal prediction has been intensely studied in the mathematics, statistics and engineering communities. Investigations of temporal predictions have their roots in the study of dynamics and of probability. Fourier series were developed to integrate the dynamics of heat [70], and the calculus of probability was developed to predict uncertain future outcomes conditioned on past evidence [127, 140]. Theses foundational ideas were used to address signal processing problems during the Second World War [121, 223]. Optimum mean squared error prediction was applied to time series whose future is only incompletely determined by their past. With the development of information theory, prediction took an new role for enabling communication [188]. Linear prediction became a central element of statistical signal processing with the Kalman filter [112]. In the meantime meteorology and fluid dynamics developed physics based prediction algorithms [172]. Stochastic forecasting methods also became central to quantitative finance [13, 141, 26]. Motion compensated video coding uses temporal prediction for data compression and remains a core element of image processing [222].

The fields of cognitive science, machine learning and computational neuroscience have also been concerned with the problem of temporal prediction since their inception. Many aspects of human cognition can be described as the use of mental models to predict future events [45]. Recurrent Neural Networks were developed and trained for next letter prediction in text, using an internal state to integrate information from the past and to extrapolate [108, 61]. Today transformer networks based on a "self-attention" architecture are widely used. These models are simply trained to auto-regressively predict the next token on large text corpora—demonstrating that temporal prediction provides a powerful learning signal [2].

But biological organisms do not passively predict their environment; instead, they actively sample, explore, and try to influence it. The field of optimal control builds methods for predicting the consequences of one's actions. Model predictive control uses forward models to predict the next state of the world given the current state and action [33, 227]. The neural circuits that compute such predictions are actively being studied [116, 185]. Predictive representations have also become central to the field of reinforcement learning [48]. But learning by acting can be costly and dangerous, so organisms should learn as much as possible through observation alone.

Pure prediction can be brittle and miss transient events or fail to generalize when the data distribution shifts. Indeed no prediction is perfect and in practice prediction methods should strive both for accuracy and well as expediency. Predictions are only useful if they can be computed quickly under resource constraints and help inform risky decisions.

Modeling biological vision. Our visual system generates a remarkably accurate and stable interpretation of the world, allowing us to make temporal predictions and adapt to constantly changing circumstances. Computing visual predictions feels effortless, but it is an impressive computational feat: the predictable structure of visual information is hidden and obfuscated in high-dimensional, dynamic and noisy measurements. What are the representations that underlie these capabilities? Some principles will help answering this question. Neurons are selective for environmental features, and these selectivies are constructed by cascades of simple transformations. Our perceptual capabilities are enabled—and limited—by the activity of neurons, their noise properties, as well as prior information. These neural selectivities have to be learned mostly from observation, *i.e.*, unsupervised. Building on those general principles, two broad normative frameworks have been used to explain the neural computations that give rise to perception: coding efficiency and optimal inference.

The theory of efficient coding proposes that sensory systems allocate resources efficiently to capture information about inputs that are likely to occur [10, 19, 196]. Coding efficiency maximizes information about stimuli in responses, subject to constraints (*e.g.*, number of neurons, spike rate, metabolic costs, wiring length). But pure coding efficiency is blind to the behavioral use of visual information. The temporal prediction theory argues that the visual system should preferentially extract the aspects of the signal that carry predictive information, and that this predictive information should be represented in a format that facilitate temporal prediction. In particular sensory systems should decompose the dynamics of the signal and extract the causal factors of variation that are useful for prediction—factorizing the underlying causes of visual measurements (*e.g.*, reflectance/illumination, identity/expression). Efficient coding and temporal prediction have already proven to be useful guides in explaining many aspects of early vision [204].

The theory of optimal inference proposes that perception is a best guess as to what is in the world, given current sensory responses and prior experience [213]. Visual information should be represented in a format that facilitates decoding: visual processing should isolate task-specific information and reformat it for downstream inference (eventually, for motor action). Probabilistic estimation provides the mathematical tools for modeling this inference process [104, 192]. As a result, optimal inference is fundamental to modern accounts of perception and action, and it also forms the basis of probabilistic machine learning [119, 134]. But optimal inference does not define which quantity is of interest and one needs to assume upfront what to estimate or decide about. The temporal prediction theory argues that the future is the relevant object for optimal inference. Furthermore, prediction at longer time horizons promotes the extraction and storage of stable information—providing a powerful signal for representation learning.

In summary, temporal prediction subsumes aspects of both coding efficiency and optimal inference while addressing some of their limitations. By unifying these two frameworks, temporal prediction avoids the common, but somewhat arbitrary, division into encoding and decoding models. In the next section, we will discuss how to represent and analyze properties of visual scenes. We emphasize the importance of natural scene statistics and of visual motion. We then discuss existing approaches and outline desiderata for representation learning based on temporal prediction.

1.2 Representing visual signals

Making temporal prediction is only possible because the world is lawful and because visual signals are highly structured. In fact, visual signals only differ from noise in so far as they are redundant. This redundancy should be exploited by sensory systems to guide the transformation of sensory messages and to drive unsupervised learning [18]. This suggests a strategy: study the empirical structure of natural scenes and design mathematical models that exploit this structure—such models can then be trained on visual signals and compared to biological systems.

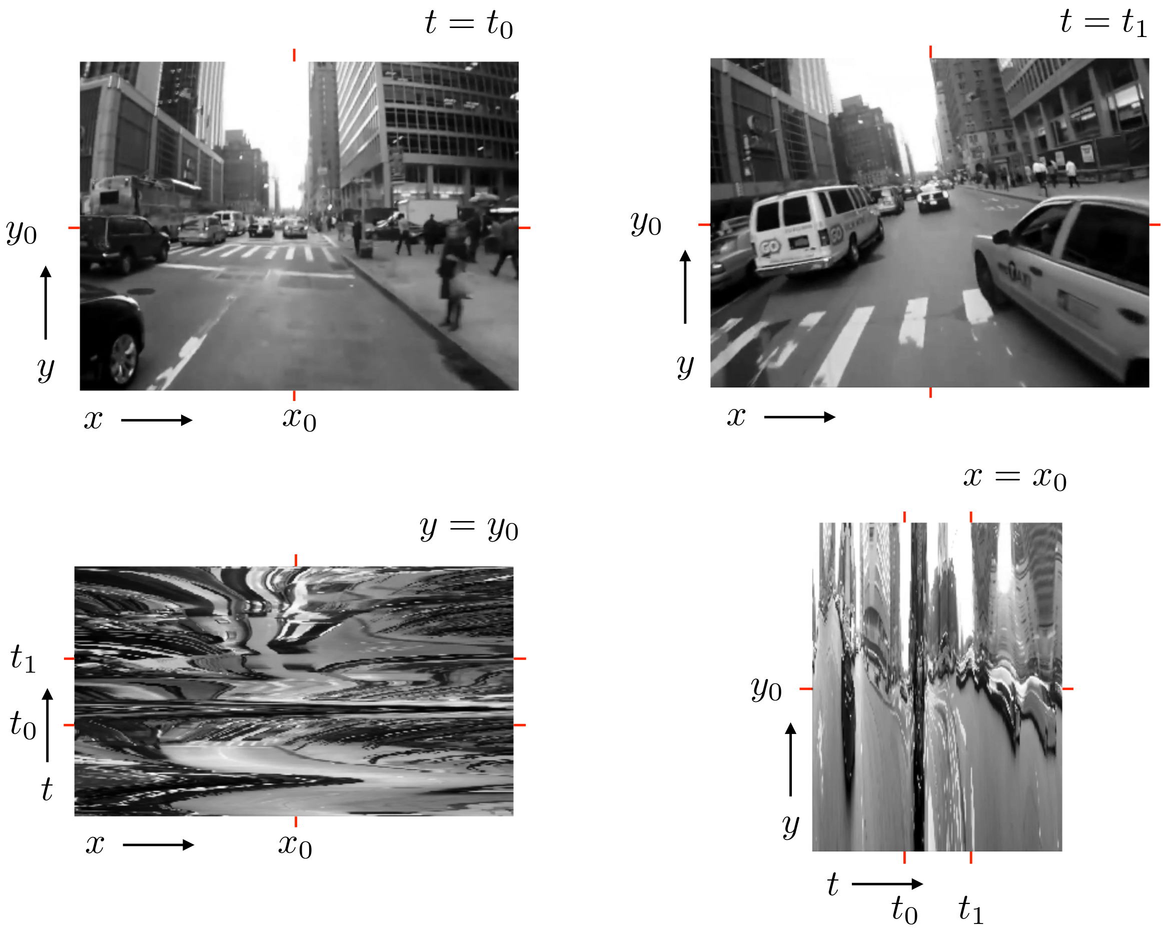


Figure 1.1: Space-time slices through an image sequence. The image sequence is an array, I[x, y, t], of size 720 × 540 × 410 pixels, measured from the point of view of a cyclist biking through traffic in Manhattan. The video was sampled at 30 frames per second. **Top.** The top two frames show slices in the x-y plane (snapshots at instants t_0 and t_1). The red tick marks indicate mid-width and mid-height and correspond to the bottom two space-time slices. **Bottom.** The bottom two frames are slices in the x-t and t-y plane, taken through the mid-height axis and mid-width axis respectively. The red tick marks along the time axis indicate instants t_0 and t_1, corresponding to the top two frames. Observe that features of the signal are qualitatively different in space and in time. Spatio-temporal continuity arises from self-motion of the observer, as well as from the motion of objects in the world. Natural visual scenes consist of multiple objects that interact in complex but predictable ways. Temporal dynamics reveal some aspects of visual signals that are not apparent on static images alone.

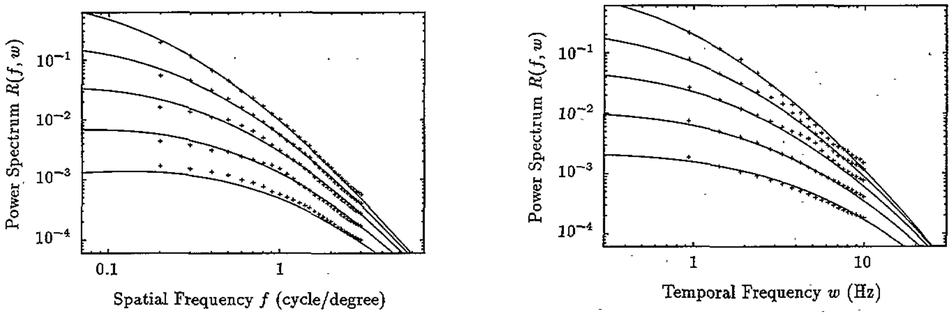


Figure 1.2: Spatio-temporal power spectrum of natural movies. Left. Joint spatio-temporal power spectrum shown as a function of spatial frequency for different temporal frequencies (1.4, 2.3, 3.8, 6, and 10 Hz, from top to bottom). **Right.** Same data, replotted as a function of temporal frequency for different spatial frequencies (0.3, 0.5, 0.8, 1.3, and 2.1 cy/deg., from top to bottom). Solid lines indicate model fits according to a power-law distribution of object velocities (reproduced from [54]).

1.2.1 Natural scene statistics

Images only occupy a small fraction of the full pixel space, a region informally referred to as the "image manifold". But our sensory systems do not process images in a random order, and visual input are not independent samples from this manifold. Instead, our eyes move several times per second, and visual signals are collected during the short fixation periods between these saccadic eye movements. These brief glimpses can be thought of as short temporal trajectories in the space of possible images—like "hairs" on the image manifold. For simplicity, we will only consider short videos captured with handheld cameras, *i.e.*, not gaze aligned. Such short videos are temporally discretized into image sequences, yet perceived as moving continuously.

Visual signals are spatio-temporally continuous and have distinctive statistics in space and in time, as illustrated in Figure 1.1. There are significant dependencies between the spatial and temporal spectra of image sequences. The slope of the spatial-frequency power spectrum becomes shallower at higher temporal frequencies, and the same is true of the temporal frequencies, as shown in Figure 1.2. This can be explained by assuming a simplified motion model for objects at different depths and power law distribution of object

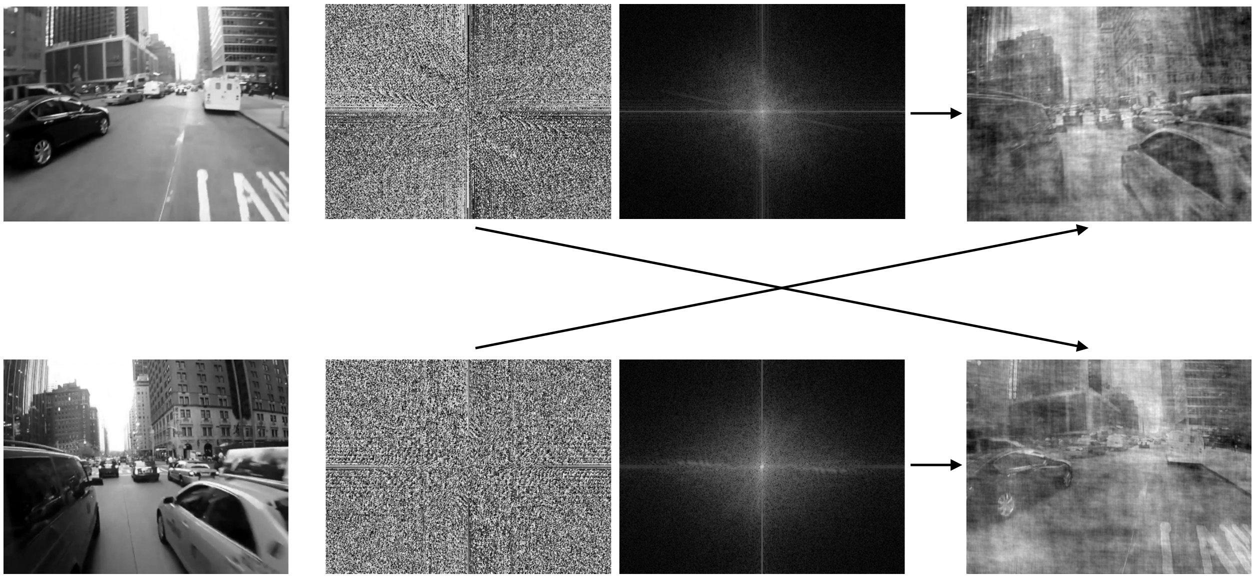


Figure 1.3: The importance of phase in images. Swapping the amplitude and the phase spectrum of two images. **Left.** The first and last frames from the example video displayed in Figure 1.1. **Center.** Phase spectrum and log amplitude spectrum in corresponding rows obtained by Fourier analysis. Both amplitude spectra are power laws and resemble each other. **Right.** Reconstructed images by Fourier synthesis of the swapped amplitude and phase spectra. The synthesized images look similar to the image whose phase was used in the synthesis. This reveals that Fourier phase carries perceptually important information.

motions [54].

But power spectra assume stationarity and only capture second order statistics, *i.e.*, linear Gaussian relationships, and therefore provide a weak approximation of natural images. This can easily be demonstrated by swapping Fourier amplitude and phase on a pair of images [155]. Image frequency amplitudes follow a generic power law, and this test reveals the importance of phase as demonstrated in Figure 1.3.

The Fourier transform used to reveal the importance of phase is computed with sinusoids that extend over the entire spatial domain. But our visual system relies on spatially localized operators: neurons in the visual system respond to properties of confined regions of visual space, *i.e.*, their receptive are spatially localized. The steerable pyramid transform can be thought of as a localized variant of the Fourier transform. This multi-scale transform decom-

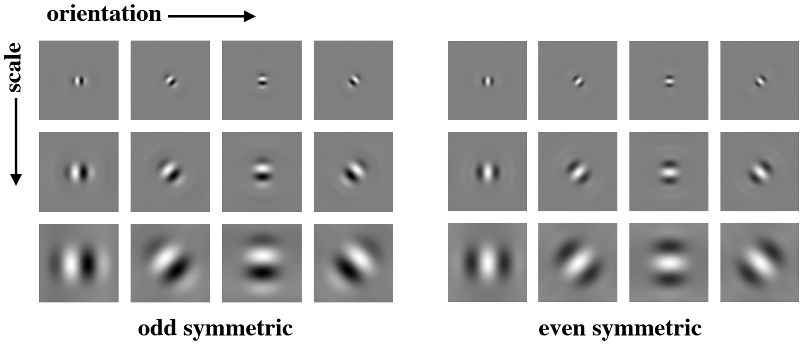


Figure 1.4: Steerable pyramid filters. Local convolutional operators used to decompose images into bandlimited and oriented channels. The highpass and lowpass residual bands are omitted. Filters at corresponding orientation and scale form the real and imaginary part of an analytic filter. These complex valued filters can be used to measure local phase and local amplitude of images.

poses image content into oriented frequency subbands via convolution with local filters [190]. The filters can be shifted in orientation (or steered), *i.e.*, their discretization respects the underlying continuous geometry of orientations [72]. A bank of such filters is displayed in Figure 1.4 where complex analytic filters composed of an even- and an odd-symmetric part constitute Hilbert transform pairs [162].

The statistics of natural image as seen through local linear filters have highly non-Gaussian statistics. Intuitively, images consist of edges interspersed with blank regions, therefore local filter responses are concentrated around zero with the rare edge coefficients spread into long tails [34, 65]. These filter responses, or filter coefficients, are sparse and their statistics can be modeled with a Laplacian distribution. Such a statistical signature presents an opportunity for removing noise from images: additive Gaussian noise on an image remains Gaussian inside the linear transform domain, while image coefficients have sparse non-Gaussian statistics. Coring methods exploit this difference and thresholds the coefficients to remove noise, this suppresses low-amplitude values while retaining high-amplitude values [193]. Beyond their sparsity, the variance of filter responses also varies in space (*i.e.*, responses are heteroskedastic). Intuitively, this is due to changes in local image contrast,

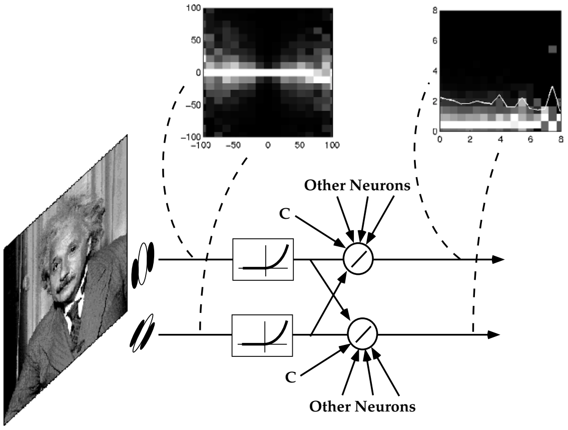


Figure 1.5: Image statistics and divisive normalization. Typical joint statistics of an image seen through two neighboring linear filters (adapted from [187]). **Left.** Filter responses are sparse and decorrelated but exhibit strong redundancy. The conditional histogram of coefficients displays a characteristic bowtie shape. In each column, pixel intensity represents the frequency of occurrence of a given pair of values. Column are independently rescaled to fill the full intensity range. Pairs of responses are gathered over all image positions. The variance of one filter's responses depends on the amplitude of responses from the other filter. **Right.** This redundancy can be removed with divisive normalization: responses are divided by a weighted sum of squared responses from neighboring filters and an additive constant. The conditional histogram of divisively normalized coefficients is flat, indicating that the variance of a filter's responses does not depend on the amplitude of a neighbor's responses.

which drives a range of filter responses, and dividing by local contrast gaussianizes the response distribution [179]. Beyond the marginal response statistics of individual filters, there are also strong local relationships between coefficients and groups of responses display conditional dependencies across location, orientation, scale, and phase. This can be revealed by considering conditional coefficient distributions [215]. Correspondingly, local gain control over local neighborhoods of coefficients further reduces redundancy, as shown in Figure 1.5. Models based on such local gain control have proven useful for image denoising [163] and can account for divisive suppression and adaptation in early sensory processing [187].

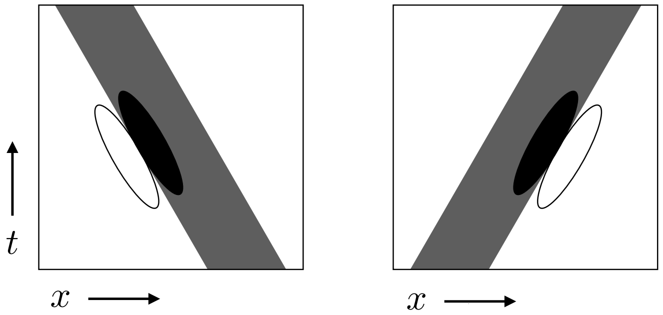


Figure 1.6: Motion is space-time orientation. Spatio-temporally oriented filters can be used to detect and measure motion. **Left.** One dimensional signal moving to the left. Motion traces a diagonal band in the two-dimensional space-time diagram. The speed of motion is given by the inverse slope of this band. An idealized filter at the corresponding orientation can measure this pattern. **Right.** Similar diagram for a signal moving to the right and a corresponding filter to measure its motion. Prediction amounts to extrapolating along local space-time orientation and should therefore be motion informed. The difference between the left and right diagram illustrates that computing such an extrapolation is highly non-linear, *i.e.*, it needs to be motion adaptive.

1.2.2 Visual motion

Humans are adept at tasks which require motion processing. Indeed, visual motion provides a rich source of information for many visual tasks such as activity recognition, object detection, material properties recognition, three dimensional pose estimation, etc. [80] Some challenging problems in computational vision are simplified by considering image sequences as opposed to static images. For example the temporal evolution of (dis)occlusions reveals object boundaries and foreground background relationships, thereby facilitating scene segmentation. And similarly, motion parallax provides a cue for optical range, which enables depth estimation. As a result, there exists a rich literature on motion processing, spanning computational theory, physiology and perception. Previous models have focused on the representation and analysis visual motion, describing physiologically plausible mechanisms for computing optic-flow, *i.e.*, estimating a single motion vector for each position in an image [191]. Those computational models were typically designed from first principles and then compared to physiological and perceptual measurements [195, 221, 233].

Visual motion also plays an important role in temporal prediction. Indeed, the motions of both the observer and objects in the scene structure the dynamics of sensory signals, allowing for partial prediction of future signals based on past ones. Motion is orientation in space-time and spatio-temporally oriented filters can be used to detect and measure it [3]. Accordingly, temporal prediction must adapt to the direction of visual motion, as illustrated by example moving patterns in Figure 1.6. Note however that prediction does not require explicitly assigning an optic-flow vector at each location of images; it should be possible to achieve more accurate visual temporal prediction with flow-free methods.

1.2.3 Representation learning

Visual representations have to be learned, mostly from observational data, by finding and exploiting redundancies and structure in signals [16]. Temporal prediction constitutes a general objective function to achieve such time-supervised visual representation learning and a variety of formulations have been considered in the literature. These formulations can be organized into three broad categories that respectively put more relative emphasis on formal elegance, empirical performance and scientific understanding.

- **Information bottleneck formulation.** This beautiful and abstract theory argues that sensory systems should extract "predictive information" [207, 25]. The amount of predictive information in a representation is measured by mutual information with future signals. The information bottleneck formulation states that representations should be maximally informative about future sensory signals while being minimally informative about past ones. This theory uses time to define a notion of relevance: sensory signals are only relevant if they pertain to the future. As such, the information bottleneck provides a statistical optimality criterion for representations; yet, it can not specify good representations because mutual information is blind to invertible mappings. Moreover, the information bottleneck formulation is computationally impractical: estimating mutual information in high dimensions is intractable, and extracting predictive information remains an open problem. As a result, it is challenging to relate the information bottleneck formulation to biological or behavioral measurements. The linear Gaussian case is a notable exception to this difficulty [39]. This solution was used to estimate the information content of a population of retinal neurons about the position of a moving bar at different temporal offsets. The information contained about the future was found to be as high as possible, given the amount of information contained about the past [157].

- **Predictive coding algorithm.** This hierarchical probabilistic method proposes a procedure for integrating top-down predictions with bottom-up errors [168]. Predictions are subtracted from feed-forward input and only residuals are signaled to the next stage of processing. This approach is inspired by video coding standards such as motion compensation where the encoder only sends residuals. The initial theory was restricted to a Kalman filter, which is a linear Gaussian model that does not correspond well to image statistics [167]. A modern deep neural network implementation of the predictive coding algorithm was trained on video data and shown to recover some qualitative aspects of biological vision [131, 132]. But the proposed architecture is complex, and is composed of stacked convolutional LSTM modules with multiple losses, making it difficult to interpret or relate to neural computation.

- **Temporal straightening hypothesis.** This functional model of visual representations proposes a computational mechanism for temporal prediction. It states that primate visual representations support prediction by "straightening" the temporal trajectories of naturally-occurring input [92]. This idea is illustrated on Figure 1.7. The temporal straightening hypothesis has received indirect psychophysical and physiolog-

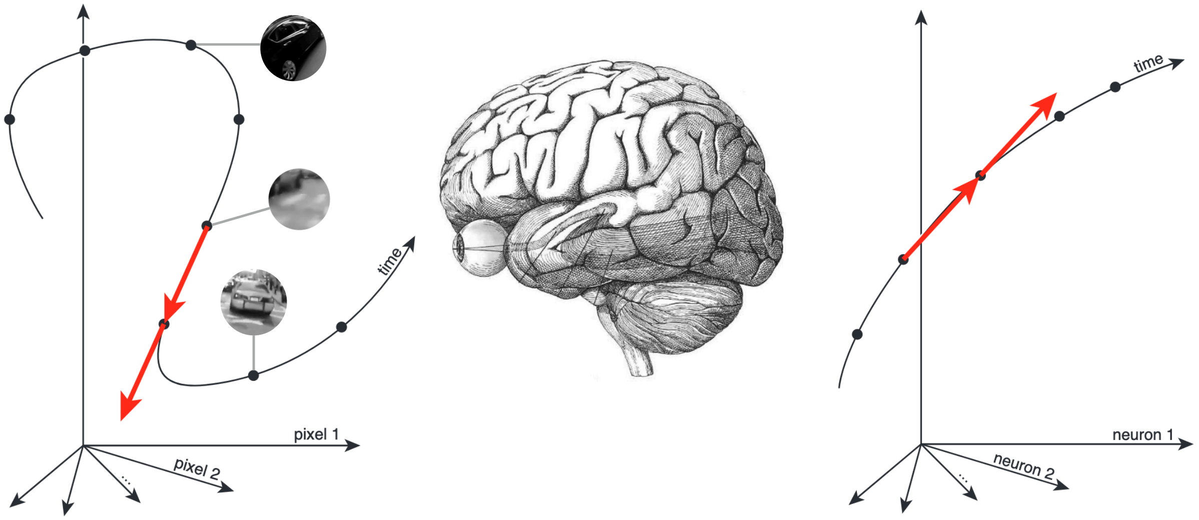


Figure 1.7: The temporal straightening hypothesis. Left. Image sequences follow highly curved trajectories in signal space, and those are difficult to predict. In particular linear extrapolation fails: the second red arrow lands far from the next image in the trajectory. **Right.** The visual system is hypothesized to represent sensory measurements in a form that simplifies temporal prediction by straightening these trajectories. Temporal prediction is reduced to linear extrapolation: the red arrows lands close to the next image in the trajectory.

ical support [93]. Building on this work, we use temporal straightening as an objective function to design and learn visual representations.

Desiderata. What do we want out of neural models of visual temporal prediction? There are three criteria that drive our work: simplicity, usability, and validity. Simple models enable understanding fundamental principles, and success in this dimension is quantified by how much we are learning by considering a particular model or theory. Usable models serve to build practical tools, and their quality is simply measured by how well they work compared to other approaches. Finally, valid models explain natural phenomena, and empirical evidence should be the eventual arbitrator between models that seek to capture the phenomena of interest.

1.3 Thesis overview

In this thesis, we will study some key aspects of the visual temporal prediction problem including representation, estimation and modeling. Our goal is to develop a unified framework for efficient and optimal predictions in neural systems. We will emphasize representations that i) are based on principles from harmonic analysis and probability theory and ii) can plausibly be mapped onto brain architecture and help understand visual perception. We will first describe methods for computing optimal temporal predictions and then assess how close to such an optimum neural circuits operate.

What is a good representation? In chapter 2, we study the empirical structure of dynamic visual scenes and propose a mathematical theory for using that structure. Abstractly, the method discovers and exploits symmetries in visual signals in order to enable temporal prediction. We demonstrate that reformatting the signal into a representation where positional displacements are explicit simplifies temporal prediction. We design a flow-free architecture that scales to the prediction of natural videos while remaining interpretable. This architecture involves a multiplicative computation which factorizes form and motion.

What is a good prediction? In chapter 3, we develop an empirical Bayes approach to visual temporal prediction. We describe a framework for estimating the uncertain future of high-dimensional image sequences and use it to perform statistical inference. We build a conditional denoising network that implicitly contains information about the posterior distribution of probable next frames given past observations. We make this information explicit by sampling probable next frames. We demonstrate that this approach automatically handles occlusion boundaries and that it adaptively combines cues.

What is the visual system computing? In the final chapter (chapter 4), we discuss the relevance of the deterministic calculation and structure proposed in the polar prediction model from chapter 2 to biological vision. We describe how cascaded neural transformations can be used to compute future predictions. We derive a model for the visual system that is based on the hypothesis that neurons should extract and exploit signal structures that can be modeled as a transformation group. We reformulate polar prediction as a model for early visual processing with local gain control, orientation and direction selectivity. We then outline a hierarchical extension to this model. Our eventual goal is to also map the stochastic solution from Chapter 3 onto physiology and perception.

Chapter 2

Representing visual transformations

"Our sensation can not give us the notion of space. That notion is built up by the mind from elements which pre-exist in it, and external experience is simply the occasion for its exercising this power, or at most a means of best exercising it. Sensations by themselves have no spatial character. [...] We have within us, in a potential form, a certain number of models of groups, and experience merely assists us in discovering which of these models departs least from reality."

- Henri Poincaré (1898), *On the Foundations of Geometry*

The fundamental problem of vision can be framed as that of representing images in a form that facilitates performing visual tasks, be they estimation, recognition, or motor action. Perhaps the most general "task" is temporal prediction, which has been proposed as a fundamental goal for unsupervised learning of visual representations [69]. Previous research along these lines has generally focused on estimating stable representations rather than using them to predict: for example, extracting slow features [225], or finding sparse codes that have slow amplitudes and phase changes [35].

In video processing and computer vision, a common strategy for temporal prediction is to first estimate local translational motion, and then (assuming no acceleration) use this to warp and/or copy previous content to predict the next frame. Such motion compensation is a fundamental component in video compression schemes as MPEG [222]. These video coding standards are the result of decades of engineering effort [60], and have enabled reliable and efficient digital video communication that is now commonplace. But motion estimation is a difficult nonlinear problem, and existing methods fail in regions where temporal evolution is not translational and smooth: for example, expanding or rotating motions, discontinuous motion at occlusion boundaries, or mixtures of motion arising from semi-transparent surfaces (*e.g.*, viewing the world through a dirty pane of glass). In compression schemes, these failures of motion estimation lead to prediction errors, which must then be adjusted by sending additional corrective bits.

Human perception does not seem to suffer from such failures—subjectively, we can anticipate the time-evolution of visual input even in the vicinity of these commonly occurring non-translational changes. In fact, those changes are often highly informative, revealing object boundaries, and providing ordinal depth and shape cues and other information about the visual scene. This suggests that the human visual system uses a different strategy—perhaps bypassing altogether the explicit estimation of local motion—to represent and predict evolving visual input.

In this chapter, we will first study the empirical structure of dynamic visual scenes and expose limitations in existing approaches to visual temporal prediction. We then propose a self-supervised representation-learning framework that extracts and exploits the regularities of natural videos to compute accurate predictions. We formulate an architecture for representing visual transformations and optimize its parameters via next-frame prediction. This polar architecture linearizes the typical transformations occurring in image sequences and is motivated by the Fourier shift theorem and its group-theoretic generalization. Through controlled experiments, we demonstrate that this approach can discover the representation of simple transformation groups acting in data. We show that, when trained on natural video datasets, this framework achieves better prediction performance than traditional motion compensation. We also demonstrate that it rivals conventional deep networks, while maintaining interpretability and speed. We end this chapter by highlighting an intriguing relationship between polar prediction and the "self-attention" block used in transformer architectures.

2.1 Dynamic visual scenes

There exists a strong relationship between visual representations and the structure of images (see discussion in section 1.2.1). Although the visual system is exposed to a stream of signals, the relation of visual representations to dynamic scenes has received less attention. In this section we focus on describing the statistics of dynamic visual scenes and ask: what makes visual temporal prediction feasible?

2.1.1 Failures of linear predictors

Linear extrapolation of images in the pixel domain fails to produce accurate predictions and instead results in double exposure images. All applications of linear extrapolation to linear representations of images suffer from the same issue.

Linear regression is a standard method for computing temporal predictions [232, 136]. Although dynamics may not be linear, they can be locally approximated with a linear solution. Such linear least squares predictions can be computed efficiently and form the basis of many engineering applications [84]. For each short segment of data, linear prediction weights are calculated, and then frequently updated to reflect the changing dynamics of the signal (*e.g.*, every 100 ms for speech applications). Unfortunately this approach is not biologically realistic as it relies on very fast weight updates while synaptic plasticity operates at a slower timescale.

Slow feature analysis seeks a signal representation such that coefficients evolve slowly in time [225]. Slowness is equivalent to straightness of the temporal difference of these coefficients. This method is typically applied after the data is lifted into non-linear features using a kernel. In that non-linear space, the slowness objective can be reduced to a generalized eigenvalue problem, but it leaves the questions of choosing an appropriate kernel untouched.

A related approach reformulates nonlinear dynamical systems as infinite dimensional linear dynamics and aims to discover a finite dimensional approximation of the corresponding Koopman operator [145]. This approach, originating in the fluid mechanics literature, can be thought of as a dynamical analog of the kernel trick. The idea is to lift a nonlinear system from the original state-space into a higher dimensional representation space where its dynamics can be captured by a fixed matrix. This formalism has inspired a line of work in machine learning where parameterized auto-encoders approximately linearize a system's dynamics in the latent space. Such networks are typically trained by minimizing a temporal prediction error [11]. Extensions have been proposed that aim to adapt the dynamics matrix to input velocity by introducing an auxiliary network, making non-linear temporal predictions in the latent space [133]. Unfortunately, learning a coordinate systems that linearizes dynamics is very challenging and these methods have not proven effective to predict natural image sequences.

2.1.2 Non-linear structure

Visual temporal prediction needs to adapt to motion and to select a direction for spatio-temporal extrapolation. As illustrated described in Figure 1.6, this is a very non-linear operation. What are appropriate representations for such motion adaptive computations? In particular, which non-linearities provide good inductive biases for temporal prediction?

Linear-Rectify-Linear. Let us start with the simplest selection mechanism: thresholding, which suppresses negative coefficients and leaves positive coefficients untouched. We consider a network that maps the previous five frames ($c_t = [x_t, \dots, x_{t-4}]$) to a predicted next-frame ($\hat{x}_{t+1}$). The network is composed of two layers: a linear analysis, followed by rectification, and a linear synthesis (or LRL cascade). The analysis contains K spatio-temporal convolutional filters (a_k of size $5 \times 7 \times 7$) and additive biases (scalars α_k). The thresholding nonlinearity, ρ, is implemented with a ReLU, *i.e.*, $\rho(.) = \max(., 0)$ (such rectified linear units are commonly used in machine learning). The synthesis contains K spatial convolutional filters (b_k of size 7×7). The network parameters ($w = \{a_k, \alpha_k, b_k\}_{k=1}^{K}$) are adjusted via stochastic gradient descent to minimize the average squared prediction error:

$$\min_{w} \sum_{t=1}^{T} ||x_{t+1} - \hat{x}_{t+1}(c_t)||^2, \text{ where } \hat{x}_{t+1}(c_t) = \sum_{k=1}^{K} B_k \rho(A_k^\top c_t + \alpha_k), \tag{2.1}$$

where A_k (respectively B_k) is a convolution matrix with the filter a_k (respectively b_k). It has been argued that such an architecture learns features that resemble the receptive fields of neurons in the early stages of the primate visual system [197]. Such example filters are displayed in Figure 2.1. But the LRL cascade has limited expressivity and this method produces inaccurate predictions, even when the number of channels K is large. This is because thresholding is an inefficient non-linearity for temporal extrapolation of image structure.

Linear-Gain-Linear. Let us now consider a slightly more sophisticated architecture that aims to explicitly use visual motion for computing temporal predictions. We consider the same least squares objective as in equation 2.1, only changing the non-linearity of the predictor, $\hat{x}(c)$ (hereafter LGL cascade). The idea is to first estimate a distribution of probable visual motions using a nonlinear gating mechanism, and to then marginalize it out. The joint distribution of target next frame x, past frames c and visual motion v, factorizes as

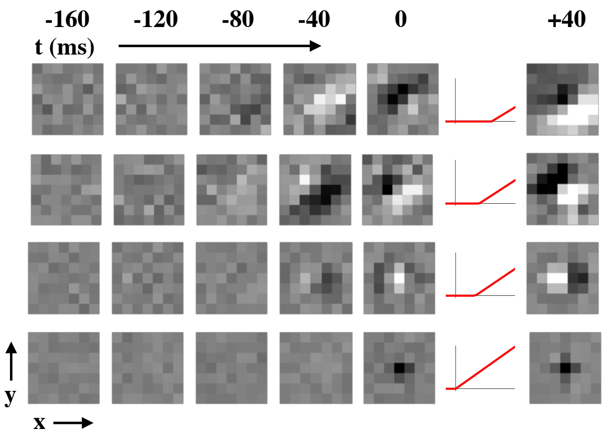


Figure 2.1: Rectified spatio-temporal filters. Four example convolutional channels in a Linear-Rectify-Linear temporal prediction model. Each row displays a space-time analysis filter on the left and a corresponding spatial synthesis filter on the right. The filters are normalized and the intensity of each channel is depicted by the slope of the rectification non-linearity in red. The spatial phase of the first two filters shifts with time, indicating that they are selective for orientation in space-time. The last filter is an identity operation and it acts as a copy-pasting mechanism on positive valued coefficients.

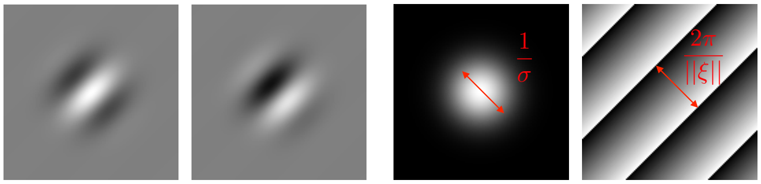
$p(x, c, v) = p(x|c, v)p(v|c)$. It is well known that optimal least squares estimation is achieved by the posterior mean, and using this factorization, the next frame posterior mean can be written as:

$$\hat{x}(c) = \int xp(x|c)dx = \int xp(x|c, v)p(v|c)dvdx. \tag{2.2}$$

We consider a discrete approximation to this integral, utilizing spatio-temporal analysis filters, a_k, spatial synthesis filters, b_k, and a softmax gating mechanism:

$$\hat{x}(c) \approx \sum_{k=1}^{K} \frac{e^{a_k^\top c}}{\sum_{j=1}^{K} e^{a_j^\top c}} (a_k^\top x) b_k. \tag{2.3}$$

The synthesis filters form a basis in which the upcoming frame is expressed. The coefficients in that basis are a product of the analysis coefficients and of the softmax gains, respectively approximating $p(x|c, v)$ and $p(v|c)$. Unfortunately this LGL cascade only offers marginal performance improvements over the LRL cascade, even when the number of channels K is



Re(ψ) Im(ψ) $|\psi|$ $\angle(\psi)$

Figure 2.2: Analytic filter in rectangular and polar coordinates. Illustrating the definition of local amplitude and local phase. **Left.** Rectangular representation of a spatially localised oscillatory pattern, or wavelet. The real and imaginary parts of the filter correspond to even and odd symmetric sinusoids. **Right.** Polar representation of the wavelet filter. The amplitude is a Gaussian envelope whose width controls frequency bandwidth. The phase is a ramp whose slope controls central frequency.

large. One important limitation of these models is that they can only handle a finite set of motion speeds, failing to exploit the distinctive spatio-temporal continuity that structures image sequences.

Local phase is fundamental. Local phase is defined with an analytical filter such as the Gabor wavelet depicted in Figure 2.2. Local phase provides a strong signal for spatio-temporal continuity as illustrated by Figure 2.3. Indeed when features move, their positional information is only implicitly available in the relative coefficient values of linear filters. Positional information can be made explicit with local phase and thereby reveal temporal stability. We give an explicit example of straightening an image sequence and revealing its temporal structure by using a local polar decomposition in Figure 2.4.

The importance of decomposing images into local phase and amplitude information has long been recognized in vision science [234]. Specifically local phase has been used in a number of machine vision and image processing applications, such as estimation of image motion [67] and disparity [68], description of image textures [162], recognition of iris patterns [47], perception of blur [218], and image compression [183, 9]. These methods all attempt to factorize visual signals—separating form and motion—to facilitate processing.

The literature on motion microscopy describes temporal processing of local phases in a

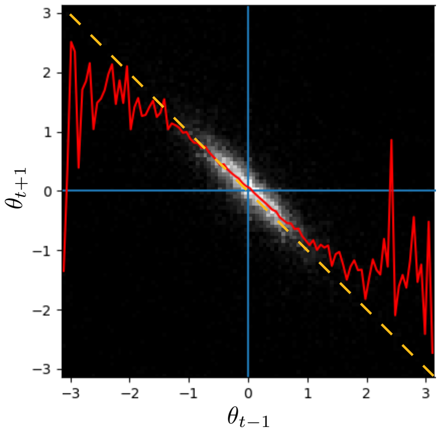


Figure 2.3: Conditional phase histogram. Statistical summary of local phase coefficients dynamics. The coefficients are computed with an analytic pair of steerable pyramid filters on a natural image sequence. The two dimensional histogram displays the phase value at the next time step (θ_{t+1}) against the phase value at the previous time step (θ_{t-1}), conditioned on the current phase value being 0 ($\theta_t = 0$). Grayscale intensity indicates frequency, and the red curve is an empirical estimation of the posterior expectation: $\mathbb{E}[\theta_{t+1}|\theta_{t-1}, \theta_t = 0]$. Statistics are collected over spatial locations and time in the image sequence. The dashed yellow line indicates minus the identity and highlights the linearity of phase relationships through time.

fixed complex-valued wavelet representation to interpolate between video frames and magnify imperceptible movements [228, 214]. This work highlights the difference between "Lagrangian approaches" and "Eulerian approaches." The former is based on optic flow: motion is computed explicitly and the frames of the video are warped according to the magnified velocity vectors. The latter bypasses the difficulties of motion estimation, and processes local phase differences, thereby separating spatial and temporal operations.

Additive vs. multiplicative methods. To end this section, we contrast linear and angular extrapolation in a wavelet representation. The comparison between these two approaches is illustrated in Figure 2.5. The notions of central frequency and of frequency bandwidth of the wavelet are depicted in Figure 2.2.

Taylor expansions are commonly used for linear approximation and extrapolation—but they are only valid locally in space. The inner product between a shifted signal x_τ with a

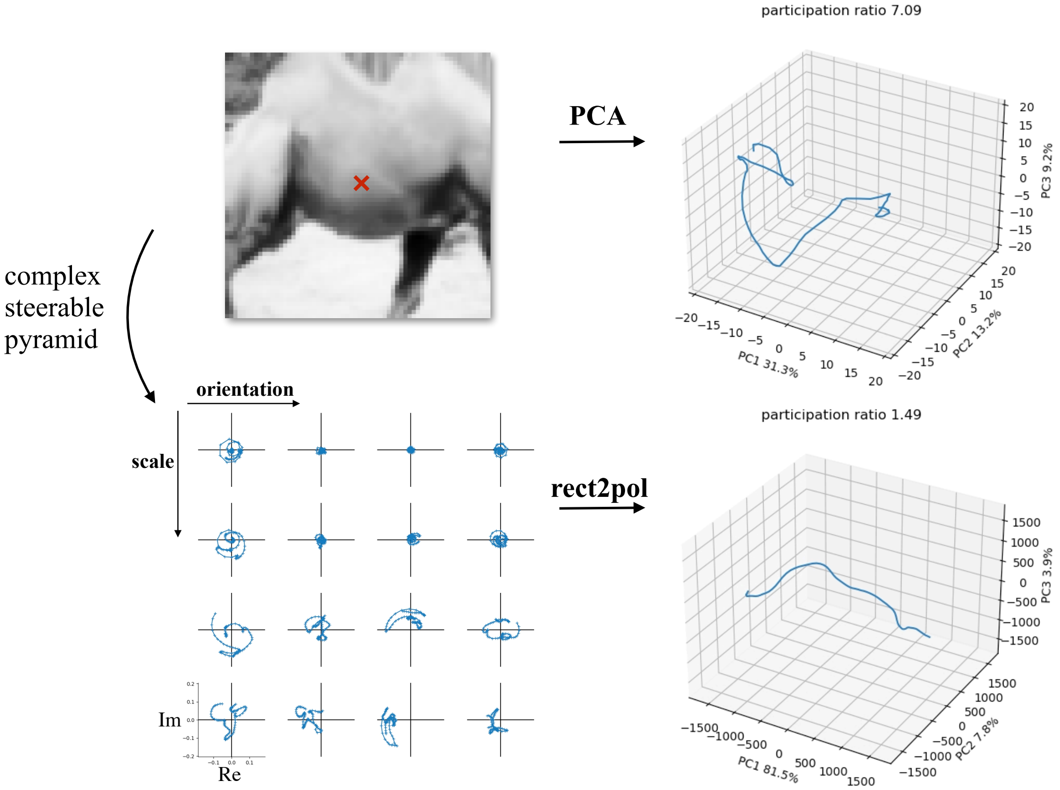


Figure 2.4: Revealing temporal structure. Straightening an image sequence via local polar decomposition. **Top left.** First frame of an image sequence involving the motion of a camel to the right and camera panning. Each image in the sequence is a point in the pixel state space. **Top right.** Projection in the first three principal components of the image trajectory in pixel state space. Notice that this trajectory is highly curved, which corresponds to the right hand side of Figure 1.7. **Bottom left.** Coefficients of steerable pyramid filters applied in the centre of the image sequence, as indicated by the red cross in the upper left. Pyramid bands are displayed with orientation on the x axis and scale on the y axis as in Figure 1.4. The two dimensional curves indicate the response to even-symmetric and odd-symmetric filters (respectively Im and Re). Notice that these trajectories are circular, suggesting transforming the responses from rectangular to polar coordinates. **Bottom right.** Projection in the first three principal components of the image trajectory in polar coefficients space. Coefficients in polar coordinates at each location were concatenated in a long vector containing amplitudes and unwrapped phases. Such vectors were collected in a matrix for all the time steps in the video. Principal component analysis is applied to this matrix. Notice that the trajectory is straightened, which is analogous to the right hand side in Figure 1.7.

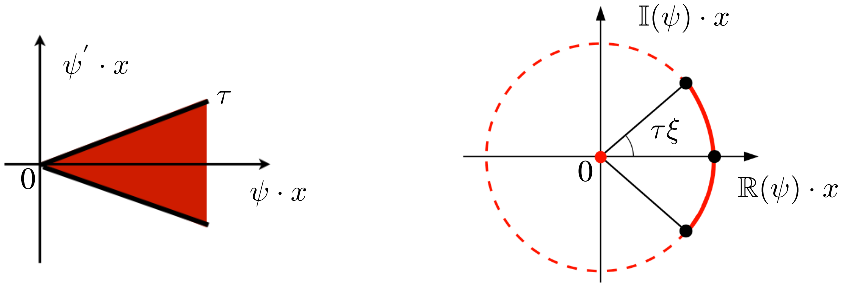


Figure 2.5: Taylor vs. Fourier extrapolation. Comparing two approaches to extrapolation. **Left.** Linear extrapolation is valid locally in space and is exact for ramps. **Right.** Angular extrapolation is valid locally in frequency and is exact for waves.

wavelet ψ, can be approximated with the inner product of the original signal x with the same wavelet by adding a term that is linear in the size of the shift τ, provided that the shift is small relative to the central frequency of the wavelet:

$$\psi \cdot x_\tau \approx \psi \cdot x + \tau(\psi' \cdot x), \text{ if } \tau < \xi^{-1}, \tag{2.4}$$

where ψ' is the spatial derivative of the wavelet, $\widetilde{\psi}$ is its Fourier transform, and its central frequency is defined as: $\xi = \frac{1}{(2\pi)^2} \int_{[-\pi,\pi]^2} \omega |\widetilde{\psi}(\omega)|^2 \, \mathrm{d}\omega$.

Fourier methods are also commonly used for angular extrapolation—but they are only valid locally in frequency. The inner product between a shifted signal with a wavelet can be approximated with the inner product of the original signal with the wavelet phase shifted by an amount $\tau\xi$ that depends both on the size of the shift and the central frequency of the wavelet, provided that the shift is small relative to the frequency bandwidth of the wavelet:

$$\psi \cdot x_\tau \approx e^{-i\tau\xi}(\psi \cdot x), \text{ if } \tau < \sigma^{-1}, \tag{2.5}$$

where the frequency bandwidth is defined as: $\sigma^2 = \frac{1}{(2\pi)^2} \int_{[-\pi,\pi]^2} |\omega - \xi|^2 |\widetilde{\psi}(\omega)|^2 \, \mathrm{d}\omega$. In other words, local image translation is equivalent to local phase shift, and a complex wavelet representation approximately diagonalizes local translations [85]. Angular methods perform well in cases where handling large continuous shifts is important and bandlimiting the signal is possible [57]. Moreover these methods preserve the Euclidian norm of the signal, which is not true of linear methods.

We hypothesize that the spatio-temporal coherence of image sequences is revealed by local phase because of the inertia of visual transformations. Therefore a good representation should extract and exploit signal structures that can be modeled as a transformation group. The remainder of this chapter is focused on developing and testing this hypothesis.

2.2 Multiplicative mechanisms for predicting transformations

2.2.1 Base case: the Fourier shift theorem

Our approach is motivated by the well-known behavior of Fourier representations with respect to signal translation. Specifically, the complex exponentials that constitute the Fourier basis are the eigenfunctions of the translation operator, and translation of inputs produces systematic phase advances of frequency coefficients. Let $\mathbf{x} = [x_0, \dots, x_{N-1}]^\top \in \mathbb{R}^N$ be a discrete signal indexed by spatial location $n \in [0, N-1]$, and let $\widetilde{\mathbf{x}} \in \mathbb{C}^N$ be its Fourier transform indexed by $k \in [0, N-1]$. We write $\mathbf{x}^{\downarrow v}$, the circular shift of $\mathbf{x}$ by v, $x_n^{\downarrow v} = x_{n-v}$ (with translation modulo N). Let ϕ denote the primitive N^{th} root of unity, $\phi = e^{i2\pi/N}$, and let $\mathbf{F} \in \mathbb{C}^{N \times N}$ denote the Fourier matrix, $F_{nk} = \frac{1}{\sqrt{N}}\phi^{nk}$. Multiplication by the adjoint (*i.e.*, the conjugate transpose), $\mathbf{F}^*$, gives the Discrete Fourier Transform (DFT), and by $\mathbf{F}$

the inverse DFT. We can express the Fourier shift theorem[1] as: $\mathbf{x}^{\downarrow v} = \mathbf{F}\text{diag}(\boldsymbol{\phi}_v)\mathbf{F}^*\mathbf{x}$, where $\boldsymbol{\phi}_v = [\phi^0, \phi^{-v}, \phi^{-2v}, \ldots, \phi^{-(n-1)v}]^\top$ is the vector of phase shift by v. See Appendix 5 for a consolidated list of notation used throughout this work.

This relationship can be depicted in a (commutative) diagram:

$$\begin{tikzcd}
\widetilde{x}_k \arrow[r, "\text{phase shift}"'] & \phi^{-kv}\widetilde{x}_k \arrow[d, bend left, "\mathbf{F}"'] \\
x_n \arrow[u, bend left, "\mathbf{F}^*"'] \arrow[r, "\text{spatial shift}"'] & x_{n-v}
\end{tikzcd} \tag{2.6}$$

This diagram illustrates how transforming to the frequency domain renders translation a "simpler" operation: phase shift acts as rotation in each frequency subspace, *i.e.*, it is diagonalized.

Prediction via phase extrapolation. Consider a signal, the N-dimensional vector $\mathbf{x}_t$, that translates at a constant velocity v over time: $x_{n,t} = x_{n-vt,0}$. This sequence traces a highly non-linear trajectory in signal space, *i.e.*, the vector space where each dimension corresponds to the signal value at one location. In this space, linear extrapolation fails. As an example, Figure 2.6 shows a signal consisting of a sum of two sinusoidal components in one spatial dimension. Mapping the signal to the frequency domain simplifies the description. In particular, the translational motion now corresponds to circular motion of the two (complex-valued) Fourier coefficients associated with the constituent sinusoids. In polar coordinates, the trajectory of these coefficients is straight, with both phases advancing linearly (at a rate proportional to their frequency), and both amplitudes constant.

[1] Proof by substituting $m = n - v$:

$$\widetilde{x^{\downarrow v}}_k = \sum_{n=0}^{N-1} \phi^{-kn} x_{n-v} = \sum_{m=-v}^{N-1-v} \phi^{-kv}\phi^{-km} x_m = \phi^{-kv} \sum_{n=0}^{N-1} \phi^{-kn} x_n = \phi^{-kv}\widetilde{x}_k.$$

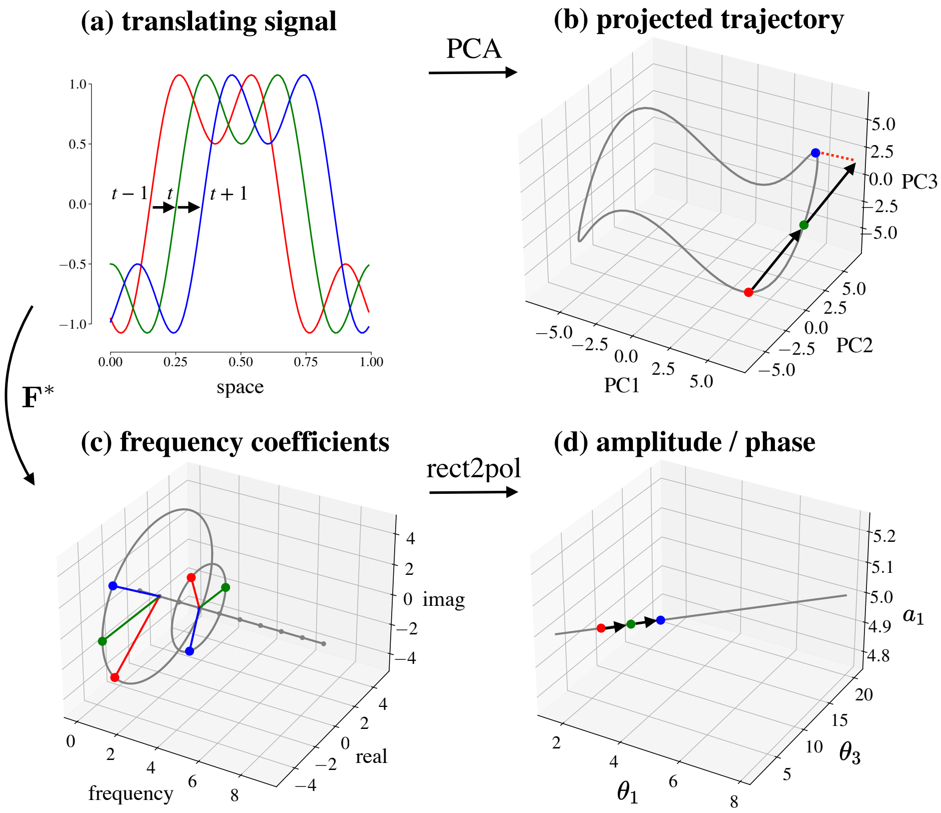


Figure 2.6: Straightening translations. (a) Three snapshots of a translating signal consisting of two superimposed sinusoidal components: $x_{n,t} = \sin(2\pi(n-t)) + \sin(2\pi 3(n-t))/2$. **(b)** Projection of the signal into the space of the top three principal components. The colored points correspond to the three snapshots in panel (a). In signal space, the temporal trajectory is highly curved—linear extrapolation fails. **(c)** Complex-valued Fourier coefficients of the signal as a function of frequency. The temporal trajectory of the frequency representation is the phase advance of each sinusoidal component. **(d)** Trajectory of one amplitude and both (unwrapped) phases components. The conversion from rectangular to polar coordinates reduces the trajectory to a straight line—which is predictable via linear extrapolation.

2.2.2 Generalization: representing transformation groups

Streaming visual signals are replete with structured transformations, such as object displacements and surface deformations. While these can not be captured by the Fourier representation, which only handles global translation, the concept of representing transformations in their eigen-basis generalizes. Indeed, representation theory describes elements of general groups as linear transformations in vector spaces, and decomposes them into basic building blocks [86]. In fact, it is illuminating to think of classical Fourier analysis as a special case in the representation theory of compact commutative Lie groups [135].

However, the transformation groups acting in image sequences are not known a priori, and it can be difficult to give an explicit representation of general group actions. In this work, we aim to find structures that can be modeled as groups in image sequences, and we learn their corresponding representations from unlabeled data.

In harmonic analysis, the Peter-Weyl Theorem (1927) establishes the completeness of the unitary irreducible representations for compact topological groups (an irreducible representation is a subspace that is invariant to group action and that can not be further decomposed). Furthermore, every compact Lie group admits a faithful (*i.e.*, injective) representation given by an explicit complete orthogonal basis, constructed from finite-dimensional irreducible representations [86]. Accordingly, the action of a compact Lie group can be expressed as a rotation within each irreducible representation (an example is the construction of steerable filters [72] in the computational vision literature).

In the special case of compact commutative Lie groups, the irreducible representations are one-dimensional and complex-valued (alternatively, pairs of real valued basis functions), *i.e.*, they have a toroidal topology. Note that this case corresponds to a concrete result from linear algebra: the equivalence between matrices that commute with one another, and matrices that admit a simultaneous block-diagonalization [98]. Moreover, the action of a compact commutative Lie group can be described as a change of phases in its irreducible representation. This suggests a strategy for learning a representation: seek pairs of basis functions for which phase extrapolation yields accurate prediction of upcoming images in a stream of visual signals.

Objective function. We aim to learn a representation of video frames that enables next frame prediction. Specifically, we optimize a cascade of three parameterized mappings: an analysis transform ($f_w : \mathbb{R}^N \to \mathbb{R}^M$) that maps each frame to a latent representation, a prediction in the latent space ($P_w : \mathbb{R}^M \times \mathbb{R}^M \to \mathbb{R}^M$), and a synthesis transform ($g_w : \mathbb{R}^M \to \mathbb{R}^N$) that maps the predicted latent values back to the image domain. All the parameters (w) of these mapping are learned by minimizing the average squared prediction error:

$$\min_w \frac{1}{T} \sum_t \|\mathbf{x}_{t+1} - g_w(\hat{\mathbf{z}}_{t+1})\|^2; \quad \text{where } \hat{\mathbf{z}}_{t+1} = P_w(\mathbf{z}_t, \mathbf{z}_{t-1}), \quad \text{and } \mathbf{z}_t = f_w(\mathbf{x}_t). \tag{2.7}$$

Polar prediction mechanism. We begin with the simplest case: sequences small image patches, linear analysis and synthesis transforms, and a fixed prediction mechanism. We will show later that polar prediction can be made convolutional and multiscale. With these extensions the model adapts to multiple unknown and noisy transformations that act in different spatial position of natural image sequences (see Section 2.3). Furthermore, we will also generalize the polar prediction mechanism to model biological computation (see Section 4.1).

Let the linear coefficients be computed as an inner product between the input patch and the filter weights. Specifically for $k \in [0, K]$, where K is the number of pairs of channels, we have:

$$y_{2k,t} = \mathbf{v}_{2k}^\top \mathbf{x}_t \text{ and, } y_{2k+1,t} = \mathbf{v}_{2k+1}^\top \mathbf{x}_t. \tag{2.8}$$

In order to obtain phases, coefficients are grouped in pairs, forming complex-valued coefficients: $z_{k,t} = y_{2k,t} + iy_{2k+1,t} \in \mathbb{C}$. These complex-valued coefficients can then be converted to polar coordinates: $z_{k,t} = a_{k,t} e^{i\theta_{k,t}}$.

With this notation, the polar prediction mechanism is defined as a linear phase extrapolation and amounts to calculating $\hat{z}_{k,t+1} = a_{k,t} e^{i(\theta_{k,t} + \Delta\theta_{k,t})}$, where the phase advance $\Delta\theta_{k,t}$ is equal to the phase difference over the interval from $t-1$ to t: $\Delta\theta_{k,t} = \theta_{k,t} - \theta_{k,t-1}$. This polar prediction mechanism makes two strong assumptions: there is no phase acceleration, and the amplitudes stay constant. The phase-advanced coefficients can be expressed in a more direct way, using complex arithmetic, as:

$$\hat{z}_{k,t+1} = \delta_{k,t} z_{k,t}, \text{ where } \delta_{k,t} = \frac{z_{k,t}\overline{z_{k,t-1}}}{|z_{k,t}||z_{k,t-1}|}, \tag{2.9}$$

with $\overline{z}$ and $|z|$ respectively denoting complex conjugation and complex modulus of z. Note that this polar prediction mechanism is homogeneous of degree one: it is computed as the ratio of a cubic over a quadratic.

This formulation in terms of products of complex coefficients has the benefit of handling phases implicitly, which makes phase processing computationally feasible, as previously noted in the texture modeling literature [162, 235]. Optimization over circular variables suffers from a discontinuity if one represents the variable over a finite interval (*e.g.*, $[-\pi, \pi]$). Alternatively, procedures for "unwrapping" the phase are generally unstable and sensitive to noise.

The estimated next patch is generated by applying the transposed filters to the phase advanced coefficients. We use the same weights for the encoding and decoding stages, that is to say the analysis operator is the conjugate transpose of the synthesis operator (as is the case for for the Fourier transform and its inverse). Sharing weights between analysis and synthesis transforms reduces the number of parameters and simplifies interpretation of the learned solution.

In summary, given a video dataset $X = [\mathbf{x}_1, \ldots, \mathbf{x}_T] \in \mathbb{R}^{N \times T}$, the convolutional filters of a polar prediction model are learned by minimizing the average squared prediction error:

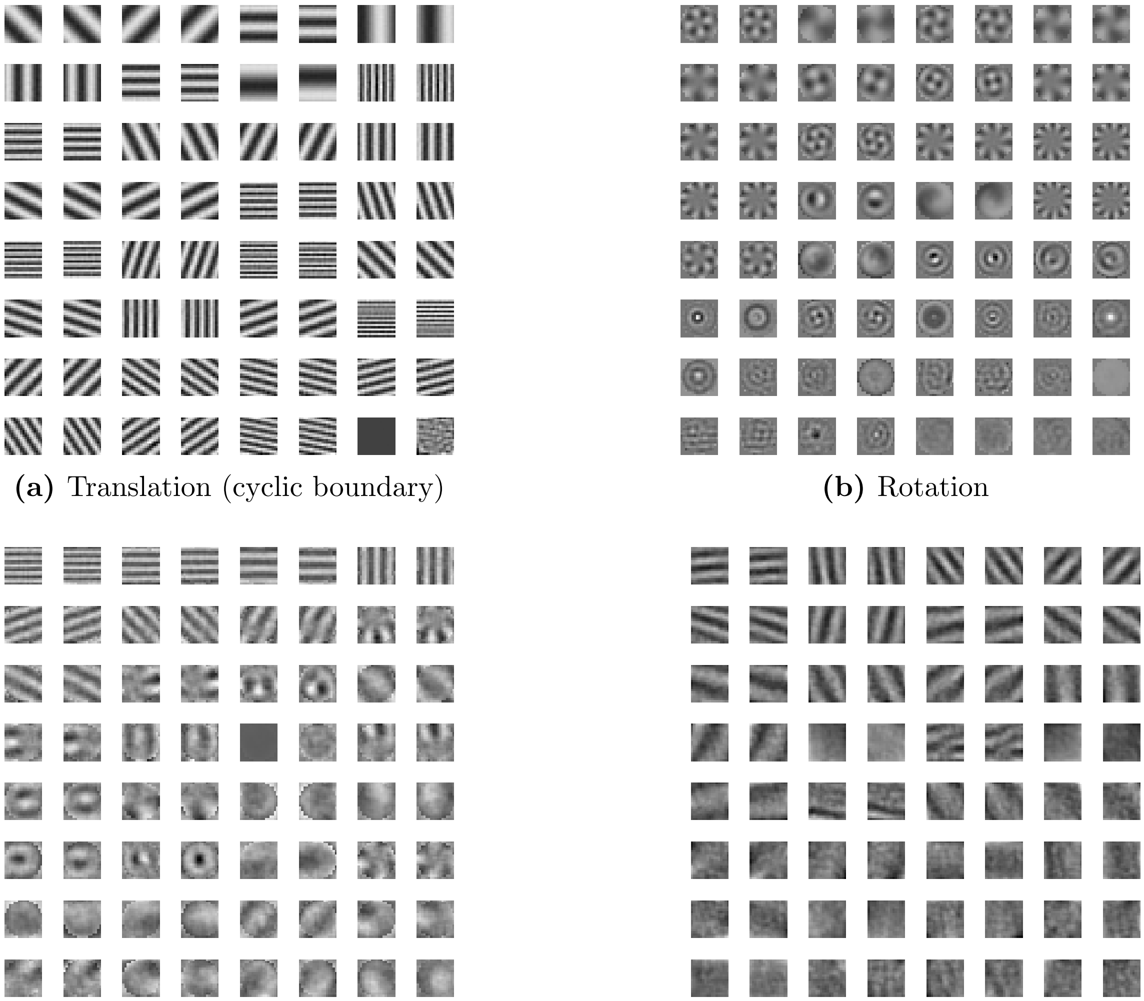
$$\min_{\mathbf{W} \in \mathbb{C}^{N \times NK}} \sum_{t=1}^{T} ||\mathbf{x}_{t+1} - \mathbf{W}\text{diag}(\boldsymbol{\delta}_t)\mathbf{W}^* \mathbf{x}_t||^2;$$

$$\text{where } \boldsymbol{\delta}_t = (\mathbf{z}_t \odot \overline{\mathbf{z}_{t-1}}) \oslash (|\mathbf{z}_t| \odot |\mathbf{z}_{t-1}|), \text{ and } \mathbf{z}_t = \mathbf{W}^* \mathbf{x}_t. \tag{2.10}$$

The columns of the matrix $\mathbf{W}$ contain the K complex-valued filters, $\mathbf{w}_k = \mathbf{v}_{2k} + i\mathbf{v}_{2k+1} \in \mathbb{C}^N$ and multiplication, division, amplitude, and complex conjugation are computed pointwise. The operations in polar coordinates is the only non-linear step used in the architecture and we therefore refer to it as the "polar predictor" model (hereafter, **PP**). This bivariate non-linear operation differs markedly from the typical (pointwise) activation function (*e.g.*, rectification) found in convolutional neural networks.

Recovery of planted symmetries. To experimentally validate our approach, we first verified that the PP model can robustly recover known symmetries in small synthetic datasets consisting of translating or rotating image patches. For these experiments, the analysis and synthesis transforms are applied to the entire patch (*i.e.*, no convolution). Learned filters for each of these cases are displayed in Figure 2.7. When trained on translating image patches, the learned filters are pairs of plane waves, shifted in phase by $\pi/2$. Similarly, when trained on rotating patches, the learned filters are conjugate pairs of circular harmonics. This demonstrates that the polar prediction model can automatically recover the representation of transformation groups acting in visual data. When the transformation is not perfectly translational (*e.g.*, translation with open boundary condition), the learned filters are localized Fourier modes. Note that when multiple kinds of transformations are acting in data (*e.g.*, mixtures of both translations and rotations), the PP model is forced to compromise on a single representation. Indeed, the phase extrapolation mechanism is adaptive but the basis in which it is computed is fixed and optimized. A more expressive model would also allow for adaptation of the basis itself.

The synthetic sequences were generated by randomly selecting 100 image patches of size 16×16 from the DAVIS dataset and transforming them multiple times to obtain a sequence of 11 frames. We applied translations or rotations and verified that PP recovers the corresponding harmonic functions: Fourier modes for translation (panel a), and disk



(a) Translation (cyclic boundary)

(b) Rotation

(c) Translation and rotation

(d) Translation (open boundary)

Figure 2.7: Learned filters on planted symmetries. Four polar predictor models were trained on synthetic image sequences consisting of translating and/or rotating image patches. In each panel, the 32 pairs of filters are displayed side by side and ordered by their norm from top to bottom. See text for discussion.

harmonics for rotation (panel b). To show that the recovery of harmonics is robust, we design two additional synthetic datasets. i) the combination of translational and rotational sequences. In this case, PP learns some filters that correspond to either transformation. But the model does not have the expressivity required to dynamically adapt the representation to either transformation (panel c); ii) generalized translation sequences: spatially sliding a square window on a large image (*i.e.*, new content creeps in and falls off at boundaries), instead of using cyclic boundary condition (*i.e.*, content wraps around the edges). In this case, PP learns localized Fourier-like modes (panel d), indicating that approximate group actions still provide meaningful training signal—but some of the filters are not structured. Notice that some of high frequency harmonics are missing. This is due to the spectral properties of image patches, which have more power at lower frequencies.

2.3 Polar prediction of natural videos

In this section, we scale the polar prediction model to natural image sequences by using convolutional multiscale operators.

2.3.1 Analysis-synthesis transforms

Local convolutional processing. When focusing on a small spatial region in an image sequence, the transformation observed as time passes can often be well approximated as a *local* translation. That is to say, in a spatial neighborhood around position n, $m \in N(n)$, we have: $x_{m,t+1} \approx x_{m-v,t}$. We can use the decomposition described for global rigid translation, replacing the Fourier transform with a learned local convolutional operator [67], processing each spatial neighborhood of the image independently and in parallel. At every position in the image, each pair of coefficients is computed as an inner product between the input and the filter weights of each pair of channels. This is analogous to the operations in Equation 2.10 but with filters repeated convolutionally at each location. Correspondingly, an estimated next frame is generated by applying the transposed convolution to the phase advanced coefficients.

Multiscale decomposition. Transformations such as translation act on spatial neighborhoods of various sizes. To account for this, the image is first expanded at multiple resolutions

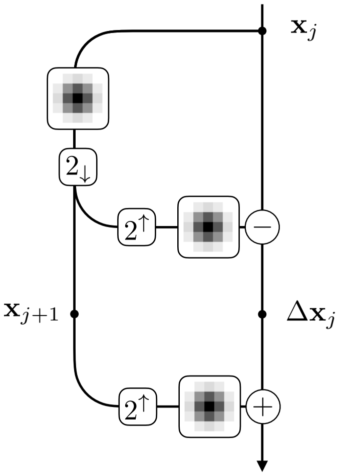


Figure 2.8: Laplacian pyramid. An image is recursively split into low frequency approximation and high frequency details. Given the initial image $\mathbf{x} = \mathbf{x}_{j=0} \in \mathbb{R}^N$, the low frequency approximation (aka. Gaussian pyramid coefficients) is computed via blurring (convolution with a fixed filter B) and downsampling ("stride" of 2, denoted $2_\downarrow$): $\mathbf{x}_j = 2_\downarrow(B \star \mathbf{x}_{j-1} \in \mathbb{R}^{2^{-j}N})$, for levels $j \in [1, J]$; and the high frequency details (aka. Laplacian pyramid coefficients) are computed via upsampling (put one zero between each sample, $2^\uparrow$) and blurring: $\Delta\mathbf{x}_j = \mathbf{x}_j - B \star (2^\uparrow \mathbf{x}_{j+1})$. These coefficients, $\{\Delta\mathbf{x}_j\}_{0\leq j<J}$, as well as the lowpass, x_J, can then be further processed. A new image is constructed recursively on these processed coefficients. First by upsampling the lowest resolution, and then by adding the corresponding details until the initial scale $j = 0$ as: $\mathbf{x}_j = B \star (2^\uparrow \mathbf{x}_{j+1}) + \Delta\mathbf{x}_j$.

in a fixed overcomplete Laplacian pyramid [34]. Then learned spatial filtering (see previous paragraph) and temporal processing (see section 2.2.2) are applied on these coefficients. Finally, the modified coefficients are recombined across scales to generate the predicted next frame (see further details in the caption of Figure 2.8).

2.3.2 Predicting natural videos

We compare our multiscale polar predictor (**mPP**) to a causal implementation of the traditional motion-compensated coding (**cMC**) approach. For each block in a frame, the coder searches for the most similar spatially displaced block in the previous frame, and communicate the displacement coordinates to allow prediction of frame content by translating blocks

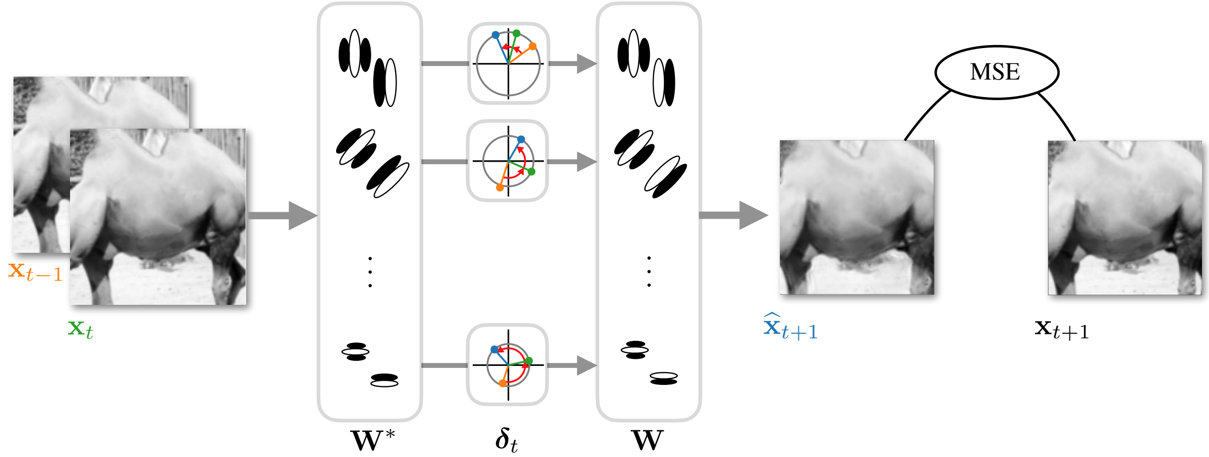


Figure 2.9: Polar Prediction Model. The previous and current images in a sequence ($\mathbf{x}_{t-1}$ and $\mathbf{x}_t$) are convolved with pairs of filters ($\mathbf{W}^*$), each yielding complex-valued coefficients. For a given spatial location in the image, the coefficients for each pair of filters are depicted in complex planes with colors corresponding to time step. The coefficients at time $t+1$ are predicted from those at times $t-1$ and t by extrapolating the phase ($\boldsymbol{\delta}_t$). These predicted coefficients are then convolved with the adjoint filters ($\mathbf{W}$) to generate a prediction of the next image in the sequence ($\hat{\mathbf{x}}_{t+1}$). This prediction is compared to the next frame ($\mathbf{x}_{t+1}$) by computing the mean squared error (MSE) and the filters are learned by minimizing this error. Notice that, at coarser scales, the coefficient amplitudes tend to be larger and the phase advance smaller, compared to finer scales.

of the (already transmitted) previous frame. We also compare to phase-extrapolation within a steerable pyramid [194], an overcomplete multi-scale decomposition into oriented channels (**SPyr**). We implemented a deep convolutional neural network predictor (**CNN**), that maps two successive observed frames to an estimate of the next frame [143]. Specifically, we use a CNN composed of 20 non-linear stages, each consisting of 64 channels, and computed with 3×3 filters without additive constants, followed by half-wave rectification. Finally, we also consider a U-net [176] which is a CNN that processes images at multiple resolutions (**U-net**). The number of non-linear stages, the number of channels and the filter size match those of the basic CNN. See Appendix 5 for a description of the architectures, datasets and training procedure.

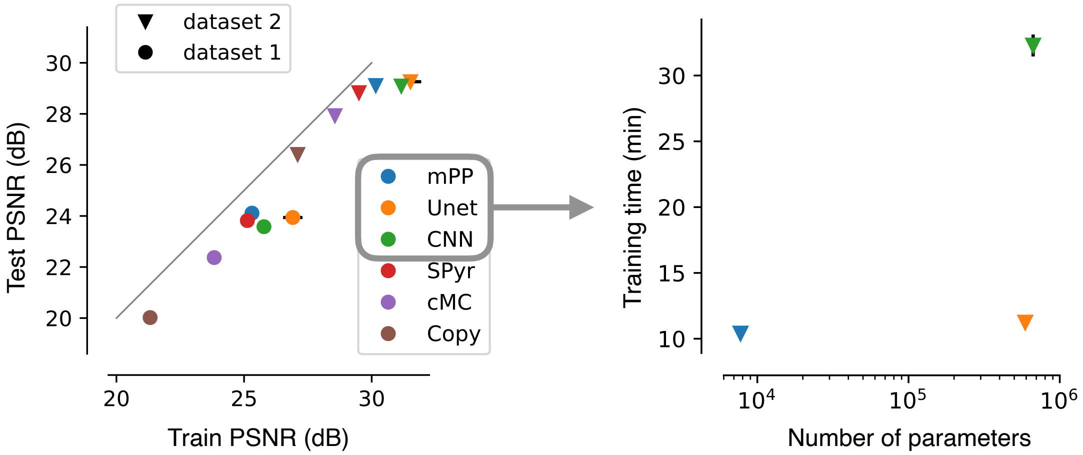


Figure 2.10: Prediction performance and computational costs. Left. Comparison of the train and test set prediction errors for several prediction algorithms, computed on the DAVIS dataset (dots) and on the VanHateren dataset (triangles). Values indicate mean PSNR and error bars represent standard deviation computed over 10 random seeds. **Right.** Comparison of the number of trainable parameters for the trainable models (with same color code as left panel). Also shown is the mean training time for 200 epochs on the VanHateren dataset, values are reported in minutes with error bars representing standard deviation. Training time is computed on a NVIDIA A100 GPU.

Prediction performance. We report performance measured by Peak Signal-to-Noise Ratio (PSNR) in decibels. The PSNR is a quality metric which expresses the logarithm of the mean squared error (MSE) in units of the signal: $\mathrm{PSNR}(x, \hat{x}) = 10 \log_{10}(\mathrm{I}^2_{\mathrm{range}}/\mathrm{MSE}(x, \hat{x}))$, where $\mathrm{I}_{\mathrm{range}}$ is the range of possible pixel values of the image. A PSNR of 0dB means that the squared peak signal and the mean squared error are equal; a PSNR of 10dB (resp. 20dB) means that the squared peak signal is 10 times (resp. 100 times) bigger than the MSE.

Prediction performance results are summarized in Figure 2.10. First, observe that the predictive algorithms considered in this study perform significantly better than baselines obtained by simply copying the last frame, causal motion compensation, or phase extrapolation in a steerable pyramid. Second, the multiscale polar predictor performs better than the convolutional neural networks on test data (both CNN and U-net are overfit). A representative example image sequence and the corresponding predictions are displayed in Figure 2.11.

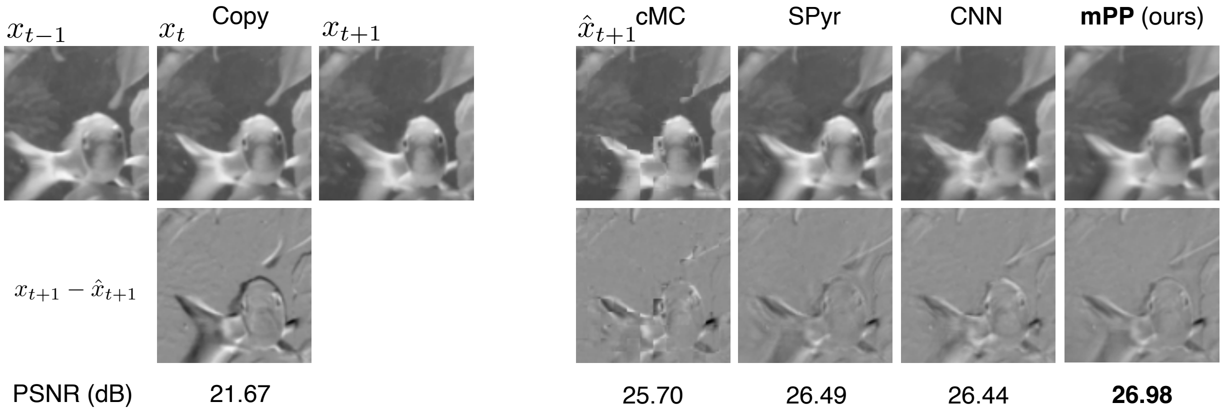


Figure 2.11: Example image sequence and predictions. Comparing prediction methods on a typical example image sequence from the DAVIS test set. **Top row.** Two observed frames (x_{t-1} and x_t) and the target next frame (x_{t+1}). Predicted next frames ($\hat{x}_{t+1}$) computed with different methods. **Bottom row.** Residual errors between the target and predicted next frame ($x_{t+1} - \hat{x}_{t+1}$) for each method, displayed in corresponding column together with performance value (measured in PSNR). All subfigures are shown on the same intensity scale. Copy: repeating the last observed frame provides a simple baseline, but it misses all the deformations of the fish. cMC: causal Motion Compensation relies on optic flow to copy-paste image patches and captures some of the movement, but it suffers from severe artefacts. SPyr: polar extrapolation in a fixed steerable pyramid representation produces a higher quality prediction, but it is inaccurate around the tail. CNN: an expressive vanilla neural network produces an accurate prediction, but is opaque and comprises 20 non-linear stages and half a million parameters. mPP: the polar prediction model prediction is most accurate (see left eye of the fish), yet comprises only a single stage and one non-linearity—thereby remaining interpretable.

Computational costs. Additional comparison of computational costs for these algorithms are also displayed in Figure 2.10. The polar predictor is lightweight and was designed as an online method that could be applied to streaming data. The polar prediction model contains roughly 30 times fewer parameters than the CNN and uses a single non-linearity, while the CNN and U-net contain 20 non-linear layers. Meanwhile PPM matches the performance of the CNN and U-net, thereby demonstrating the efficiency of the polar architecture. Notice that the U-net, although it contains about as many paramters as the CNN, is almost as fast to train as the polar predictor. Indeed most of the computation is applied to spatially downsampled coefficients. Finally, we also compare sample complexity by evaluating prediction

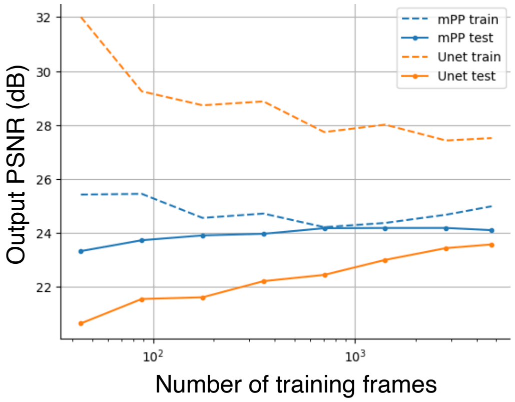


Figure 2.12: Sample complexity Prediction performance of the polar predictor and U-net measured in PSNR as a function of the size of the training set. Training performance decreases with dataset size while test performance increases. The polar prediction model is more data efficient than the U-net and achieves higher performance with small amounts of training data. Performance was computed after 200 training epochs on a subset of the DAVIS dataset.

performance of datasets of varying size. The results are reported in Figure 2.12 and show that the polar predictor is more data efficient that the U-net.

2.3.3 Auto-regressive formulation and "self-attention"

In this section, we briefly comment on the intriguing relationship between polar prediction and "self-attention." Attention layers are the key module that gets stacked in transformer networks. They are a canonical computation of modern deep learning and have found applications well beyond Natural Language Processing and language translation where it was initially developed [212]. In particular this module has proven very effective on next-token prediction tasks [2]. Yet it remains difficult to precisely describe representation power of self-attention, its benefits and limitations as compared with other canonical modules (such as convolution and rectification).

We begin by reformulating the polar prediction model in an auto-regressive form. The

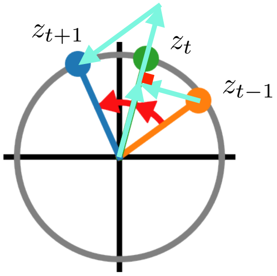


Figure 2.13: Auto-regressive polar prediction. Two equivalent computations: curved red arrows indicate vector rotation from z_t by the same angle as between z_t and z_{t-1}; cyan vectors indicate reflection of z_{t-1} along z_t, computed as the projection of z_{t-1} onto z_t, doubled, minus z_{t-1}. Both land on z_{t+1}.

polar prediction model from Equation 2.10, has the form:

$$\hat{\mathbf{x}}_{t+1} = \sum_{k=1}^{K} \mathbf{W}^k \mathbf{D}^k_{t,t-1} \mathbf{W}^{k\top} \mathbf{x}_t,$$

which is a sum of outer products, with an adaptive dynamics matrix that acts in each latent two dimensional subspace by applying a rotation to each complex-valued coefficient. This rotation can be written as a combination of past coefficients:

$$\hat{z}_{t+1} = \alpha_0 z_t + \alpha_1 z_{t-1} \text{ where } \alpha_0 = \frac{z_t \overline{z_{t-1}} + \overline{z_t} z_{t-1}}{|z_{k,t}||z_{k,t-1}|}, \text{ and } \alpha_1 = \frac{-|z_t|}{|z_{t-1}|}, \tag{2.11}$$

which corresponds to a reflection computed via projection and scaling. This operation is known as a Householder reflection in linear algebra [100] and is illustrated in Figure 2.13. This equivalence between vector rotation in a plane and reflection along that vector generalizes to arbitrary dimension.

Using this observation, and writing $\mathbf{X}_0 = [\mathbf{x}_0, \dots, \mathbf{x}_T - 1]$ and $\mathbf{X}_1 = [\mathbf{x}_1, \dots, \mathbf{x}_T]$, polar prediction can be written in an auto-regressive form:

$$\hat{\mathbf{X}}_1 = \sum_{k=1}^{K} \mathbf{W}^k (\mathbf{W}^{k\top} \mathbf{X}_0) \mathbf{A}^k_0, \tag{2.12}$$

where the matrix of auto-regressive coefficients, $\mathbf{A}_0^k \in \mathbb{R}^{T\times T}$, depends on $\mathbf{X}_0$. More explicitely, the last column of this auto-regressive matrix can be written:

$$\begin{aligned}\mathbf{A}_0^k[:, T-1] = [0, ..., 0, -\text{norm}(\mathbf{x}_t^\top \mathbf{W}^i)/\text{norm}(\mathbf{W}^i \mathbf{x}_{t-1})),\\ 2\text{normalize}(\mathbf{x}_t^\top \mathbf{W}^{k\top})\text{normalize}(\mathbf{W}^k \mathbf{x}_{t-1})]^\top \in \mathbb{R}^T,\end{aligned} \tag{2.13}$$

it contains interactions between the two dimensional coefficients of the inputs along the k^{th} basis element. This $\mathbf{A}$ matrix modulates coefficients with weights that are computed across time points. This modulation is normalized so that the whole prediction process stays homogeneous: scaling the inputs results in an equivalent scaling of the outputs.

This is reminiscent of the self-attention mechanism, which can be written:

$$\hat{\mathbf{X}}_1 = \sum_{k=1}^{K} \mathbf{W}_o^k (\mathbf{W}_v^k \mathbf{X}_0) \mathbf{A}_0^k \tag{2.14}$$

$$\mathbf{A}_0^k = \text{softmax}(\text{triu}(\mathbf{X}_0^\top \mathbf{W}_k^{i\top} \mathbf{W}_q^i \mathbf{X}_0)), \tag{2.15}$$

where the softmax is computed column-wise on the upper triangular matrix of similarities, and $\mathbf{W}_o^k$ (resp. $\mathbf{W}_v^k$, $\mathbf{W}_k^k$, $\mathbf{W}_q^k$) correspond to the k^{th} group of columns in the output matrix (resp. value, key and query matrices). Similar notational convention is often used in the machine learning literature [59]. The attention matrix in this architecture plays a role similar to that of the auto-regressive coefficients matrix in polar prediction, Eq 2.13. Here the softmax non-linearity achieves the same homogeneity property as normalization in auto-regressive polar prediction.

The relationship between auto-regressive polar prediction and self-attention offers a way to understand the implicit bias of the self-attention module, and suggests an explanation for the success of the transformer architecture: self-attention is well suited to represent and utilize the transformation groups acting in data. Related observations were made on algebraic tasks where a single self-attention layer is trained to perform discrete arithmetic operations [150, 41]. In such tasks, the weights of self-attention learn a representation of a cyclic group. It would be interesting to ask whether such circular features also underlie the performance of larger models that utilize transformers. Further analysis is required to clarify the similarities and differences between auto-regressive polar prediction and self-attention.

2.4 Discussion

We have described how the structure of image sequences is shaped by local content deformations and we introduced the polar prediction model to take advantage of this structure. By starting from mathematical fundamentals and considering an abstract formulation in terms of learning the representation of transformation groups, we formulated a framework that makes three main contributions.

Discovering group transformations. First, the polar prediction model can extract the structure implicit in sequential data: the model jointly discovers and exploits the approximate symmetries acting in image sequences. The polar prediction model handles transformations that are locally commutative and continuous. This complements the well developed group-theoretical methods in machine learning where many authors have discussed imposing known invariants [122, 44, 30]. Note that these methods learn on groups, but do not learn the group.

Symmetry discovery is a fundamental problem that has received comparatively less attention from the machine learning community. Exceptions to this include methods that adopt a Lie group formalism to factorize *invariance* and *equivariance*. Initial approaches attempted to learn group generators from dynamic signals [166], and visual representations that are invariant to geometric transformations occurring in image sequences [24]. A polar parametrization was used to identify irreducible representations in synthetic data [43]. The Lie group formalism has also been combined with sparse coding [40, 38] to model natural images as points on a latent manifold. More recently, bispectral neural networks [184] have been shown to learn image representations invariant to a given global transformation. In a related formalism, factored Boltzmann machines have been proposed to learn relational features [144]. The idea of learning a temporally straightened representation from image sequences was explored using a heuristic extrapolation mechanism: specialized "soft max-pooling" and "soft argmax-pooling" modules [82].

Notice that the polar prediction model relies on a plain temporal prediction objective. It does not promote sparsity of either amplitude or phase components and does not rely on explicit regularization. Previous work based on sparse coding also aimed to represent visual transformation and describes local image structure in polar coordinates. Specifically, a two layer complex-valued sparse coding model model was proposed for representation learning from image sequences [35]. But the discontinuity arising from selection of sparse subsets of coefficients seems at odds with the representation of continuous group actions [130]. In contrast, the polar prediction model relies on a smooth and continuous parameterization to jointly discover and exploit the transformations acting in sequential data.

A simple framework. Second, the polar prediction model achieves accurate next-frame video prediction within a principled framework and provides an interpretable alternative to opaque deep net architectures. Prediction is computed via adaptive phase extrapolation in a fixed shiftable basis. The model strictly separates spatial and temporal processing. The basis is optimized by minimizing mean squared prediction error and aims to best diagonalize video dynamics. The prediction mechanism assumes that phase evolves linearly with no acceleration and that amplitude remains unchanged. The phase relationships are exploited implicitly in a bundled computation, bypassing the instabilities and discontinuities of angular phase variables. The model applies a non-linearity independently on each complex valued coefficient.

The standard compression and denoising algorithms also rely on a similar structure. In all these signal processing tasks diagonal adaptive operators in appropriate bases offers qualitative improvements over classic linear methods. Quantizing coefficients in a discrete cosine (or wavelet) basis improves on linear compression (blurring and down-sampling). Similarly, thresholding coefficients in the basis that optimally separates signal from noise improves on Wiener filtering (convolution with an appropriately scaled low-pass filter). Our empirical results demonstrate that a related strategy is also very efficient for predicting image sequences. The standard deep networks considered here could in principle have discovered a similar solution, but did not do so in our experiments. This exemplifies a fundamental theme in computational vision: when possible, let the representation do the analysis.

Relationship to biology. Third, the non-linear components of the polar prediction model map to known physiological elements of the early visual system. In chapter 4 we will discuss the relevance of this model to understanding the nonlinear response properties of neurons in primate visual cortex. Moreover, we will treat polar prediction as a module that can be cascaded.

Unfortunately, the polar prediction model does not resolve the problem of discontinuous motion at occlusion boundaries. The model is limited to local deformations and only computes a single best guess for the next-frame. When the future frame could go in one of two directions, the model compromises and guesses the average of these two solutions. Such blurry predictions are inadequate and call for a reframing of temporal prediction as a probabilistic inference problem. This leads us to the topic of the next chapter: estimating optimal predictors.

Chapter 3

Estimating optimal predictors

> "*Instead of fearing wrong predictions, we look eagerly for them; it is only when predictions based on our present knowledge fail that probability theory leads us to fundamental new knowledge.*"
>
> - E.T. Jaynes (1985), *Maximum entropy and Bayesian methods in applied statistics*

Predicting the future is inherently uncertain and the visual system faces substantial ambiguity while anticipating sensory signals. Some of this uncertainty can be traced to measurement limitations: although remarkably accurate, our eyes only capture a fraction of the information carried by light. Moreover, our visual system inevitably discards some of this information during processing. These limitations prevent visual perception from fully resolving contingencies in the dynamics of the environment. The ambiguity caused by missing information can change outcomes and be consequential (think, for example, of the sudden direction changes in a chase between predator and prey). As a consequence our perceptual system must make assumptions and fill in those gaps. In that sense, perception is a process of inference in which incoming information is combined with internal expectations. Such unconscious inference is probabilistic in nature and must rely on information distilled from past experience. But probabilistic inference on incomplete and noisy visual measurements is computationally extremely challenging, even unfeasible in general. Yet, our visual system provides us with a reliable interpretation of the world.

In this chapter, we describe a method for estimating and sampling from the distribution of probable future frames. We first examine the high-dimensional density estimation problem in the case of the distribution of future visual signals given past observations and review some related work. We then develop an implicit method for learning, representing and utilizing an image density for temporal prediction. We show that training for resilience to noise is sufficient for embedding the density in a neural network and that this implicit density can be revealed via sampling. We highlight the case of occlusion boundaries, where the need for a probabilistic approach to prediction is most apparent, and show that the method generalizes to inference on natural image sequences.

3.1 High-dimensional probabilistic modeling

Density estimation is the mother of all unsupervised learning problems. The simplest method for estimating the distribution of a signal from data is to build a histogram. First define a grid to partition the data space into cells, then assemble a dataset, and count the number of data points that lie in each of those cells to get an estimated frequency. Unfortunately this method is cursed by dimensionality: the number of cells grows exponentially with data dimensionality. For high-dimensional data like images, this exponential dependency in the dimensionality renders the histogram based approach infeasible. For example, in the case of small gray-scale images of size 32×32 pixels with intensity sampled on a standard an 8-bit grid (corresponding to integer values in 0-255), the number of cells is 256^{1024}. This very large number is far beyond the size of any feasible dataset. In general, the statistical difficulty of density estimation obstructs probabilistic modeling, where inference and sampling require not only estimation but also integration over high-dimensional densities. Making assumptions is therefore necessary for probabilistic inference on high-dimensional data, and we need principles to guide these choices.

3.1.1 Coping with incomplete information

Many researchers have proposed methods to tackle this statistical challenge in the context of video prediction [156]. We will review three key ingredients of the problem: the need for a probabilistic formulation, the objective function (explicit vs. implicit probability modeling), and the choice of architecture.

The need for a probabilistic formulation. In ambiguous situations where multiple next frames are probable, single deterministic guesses are inadequate. Unfortunately this situation is far from marginal and image measurements are always incomplete, making them insufficient to fully specify the future. The next frame in an image sequence is a single event with several possible outcomes, its distribution is therefore multi-modal. This uncertainty can be made more visible by predicting further out into the future, as illustrated in Figure 3.1. A trained deep network predictor is applied recursively: starting with two images from the test set, and predicting the third frame. The network then predicts the following frame by processing the second and third frame, *i.e.*, its previous output is fed back as an input. After a few steps of this recursive procedure, the predictions collapse to a blurry uninformative image. The prediction network compromises, instead of deciding, on ambiguous features. Indeed, a network trained to minimize prediction mean square error approximates the posterior mean. As a result it averages over all possible next frames. But the manifold of natural images does not form a convex set, and averaging pushes the prediction off the image manifold, resulting in unnatural predictions. This weakness is common to all the deterministic methods considered

Figure 3.1: Collapse of recursive deterministic prediction. First two frames of a test image sequence, followed by predictions from a trained CNN. Each predicted frame is the result of processing the last two frames in the sequence, *i.e.*, the predictor is applied recursively. Predictions become blurry after a few steps and collapse to an uninformative smooth image.

in Chapter 3.

Although no perfect prediction is possible, an optimal method should aim to make full use of all available information while remaining noncommittal about the rest. Such a probabilistic approach requires assigning a belief to all possible next frames, which is infeasible. Instead, classical solutions assume a stationary stochastic processes, and linear Gaussian models have been used for filtering at least since Wiener [223]. In that case, mean and covariance are sufficient statistics and can readily be computed. This approach has been successfully applied to dynamical systems where the Kalman filter combines noisy measurements and linear state dynamics [112]. Such linear Gaussian methods form the bedrock of machine learning [178] but they are insufficient to tackle problems with multi-modal posterior distributions such as video prediction.

Many extensions have been proposed to handle non-linear dynamics and complex posterior distributions. One approach makes locally linear approximations to the dynamics (extended Kalman filter), unfortunately it only considers Gaussian posteriors. A general strategy for handling more complex posteriors is to make locally Gaussian approximations: with the mean set at the maximum of the posterior and the precision matrix set to the Fisher information (*i.e.*, a Laplace approximation). This construction is well suited to unimodal posterior distributions. To tackle multimodal posteriors, particle filters have been developed that track the trajectory of multiple possible samples through time. Unfortunately these methods suffer from the curse of dimensionality. The rest of this section is devoted to discussing modern machine learning solutions, which offer expressive and flexible approaches for probabilistic visual temporal prediction.

The objective function. At a high level, probabilistic methods aim to map data from the signal space into a representation that is distributed according to a Gaussian and to map representations back to the signal space. Explicit approaches to generative modeling aim to learn a proper probability density. These methods need to compute a normalization constant to ensure that the learned density function integrates to one. The standard objective for generative modeling is KL divergence between estimated and target distribution, and it is achieved by maximizing likelihood. These likelihood-based models make assumptions in order to tackle the problematic normalization constant (it is an integral over the high-dimensional data distribution). As a result these methods suffer from limitations on the family of densities that can be represented, or on the family of architectures that can be used.

- Variational Auto-Encoders [118, 171] are latent-variable models that build on principles from variational inference [216, 27] and are trained to minimize a bound on the log likelihood of the data. The idea is to substitute the inference algorithm with a learned mapping, a deep encoder network, that outputs an approximate posterior over the latent for each datapoint. Sequential variational auto-encoders have been applied to video prediction but with limited success [125, 66, 51, 12]. These methods rely on approximations and the variational bound can be quite loose; they also assume a fixed Gaussian prior in the latent space and this limits their expressivity.

- Normalizing Flows [53, 170] use the change of variable formula to compute a series of invertible mapping between the data distribution and a Gaussian distribution. This "gaussianization" approach sidesteps intractable integrals, but is restricted to architecture for which computing the change of variable is efficient (it involves a problematic log determinant term).

- Pixel autoregressive models [21, 79, 209] factorize the data distribution into a series of conditional distributions and learn those smaller conditional distributions. But these methods proceed in an arbitrary pixel raster order (adopted from the coding literature), and take more time for generating samples, which makes them inappropriate for an online temporal prediction setting.
- Prediction of discretized video token [165] forms an explicit multinomial distribution over a quantized representation of the next frame. This approach is inspired by the success of next token prediction in Natural Language Processing which demonstrated the power and generality of a simple prediction objective function. In language modeling the number of tokens is large but remains manageable: it can be explicitly modeled as a multinomial distribution and optimized with the familiar categorical cross-entropy loss that is also in common use for multi-way classification. Application to high-dimensional video data requires tokenization, for example via learned vector quantization [210] (which is optimized with a straight-through estimator, bypassing the non-differentiable quantization step). But the quantization step often occurs early in processing and produces representations that can be brittle.

The difficulties of likelihood-based models can be avoided by considering implicit models. A prominent example of this approach is given by Generative Adversarial Networks [81] which only models the generation process and have been widely used for video prediction [143, 217]. The idea is to train a generative model by optimizing it to fool a classifier that tries to distinguish between images from the training set and samples from the model. Unfortunately these models are unstable and drop modes of the data distribution. The competitive game between generator and discriminator networks is sensitive to details of the training algorithm.

The game effectively minimizes a Jensen-Shannon divergence, but fails to represent certain parts of the distribution. Nevertheless, the idea of an implicit approach remains appealing, especially if it could be made to provide some access to the desired probability distribution.

Architectural choices. Video next-frame prediction is usually cast as a sequence-to-image problem, and a central difficulty is to handle long-term temporal dependencies. Recurrent Neural Network make use of long context by updating a latent state, but are difficult to train. In practice it is common to resort to complex self-gating architectures like the Long Short-Term Memory networks [131]. Feedforward Convolutional Neural Networks on the other hand are much more stable to train, but only rely on limited temporal context—although pyramids of causal convolutions may be used [154]. Many authors have proposed more sophisticated video processing architectures that explicitly splits into motion and content streams. These methods build in some structural assumptions, for example the notion of background and foreground, or some notion of objectness. Similarly, using motion-compensation upfront suffers from the limitations of optic-flow: assigning a single displacement vector per pixel fails at occlusion boundaries, where there are multiple motions. Some researchers have designed architectural variants that aim to strike a balance between structure and flexibility, obtaining promising results in tasks such as image animation and action recognition [230, 63]. Of particular interest are architectures that exploit the multiscale nature of visual signals, in particular U-nets [176] are auto-encoders that process signals fine-to-coarse and then coarse-to-fine.

3.2 Score-based inference

Score matching estimation offers an alternative to maximum likelihood estimation that bypasses the problematic high-dimensional integral in the normalization constant [102].

Maximizing the log likelihood $\mathbb{E}_p\big[\log p_\theta(x)\big]$ is equivalent to minimizing the KL-divergence $KL(p, p_\theta) = \mathbb{E}_p\big[\log p(x) - \log p_\theta(x)\big]$, where $p(x)$ is the data distribution and $p_\theta(x)$ the density model. But optimizing the parameters of a density model by minimizing KL divergence is computationally prohibitive: it involves taking gradients of the normalization constant with respect to the parameters. Score matching eliminates the normalizing constant by replacing KL minimization with minimization of the weaker relative Fisher information:

$$\mathcal{I}(p, p_\theta) = \mathbb{E}_p\Big[\frac{1}{2}||\nabla_x \log p(x) - \nabla_x \log p_\theta(x)||^2\Big]. \tag{3.1}$$

In other words score matching minimizes the expected squared distance between the gradient of the log-density given by the model and the gradient of the log-density of the observed data. In the statistics literature, the gradient of a log-density is also know as a score function, hence the name score matching. For models in the exponential family (*i.e.*, whose energy is linear in θ), this estimator can be expressed as a simple quadratic objective in the parameters (via integration by parts). Moreover the score matching estimator is consistent and its precision depends on the smoothness of the data distribution.

This estimation method was generalized and placed in an "empirical Bayes" framework [169]. This framework is appealing because of its simplicity and generality: it provides least squares estimators given access to data and to the distortion process only. Indeed, denoising and least squares optimization can be used implicitly to learn arbitrary data distributions. In the rest of this section, we contrast this framework with other machine learning approaches, and provide an intuitive introduction to empirical Bayes, focusing on approximation and optimization.

Probabilistic modeling frameworks. The two critical differences between the empirical Bayes approach and other probabilistic machine learning approaches are i) denoising is a local operation in the space of probability distributions and ii) denoising serves both for learning

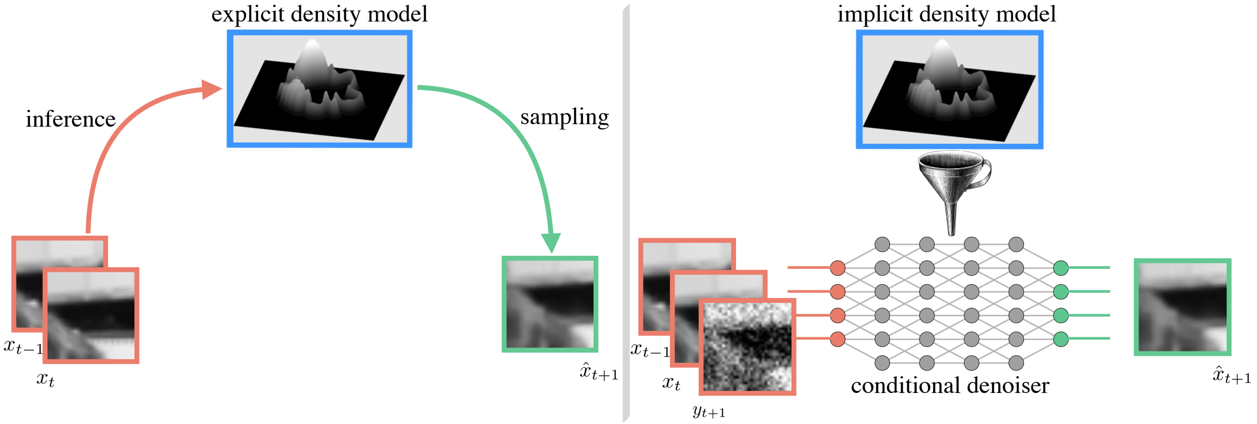


Figure 3.2: Probabilistic modeling frameworks. Contrasting implicit and an explicit approaches to probabilistic modeling. In both cases, a model transforms frequencies estimated from past measurements into beliefs assigned to possible future measurements. **Left.** Traditional framework: given training data, perform inference in a probabilistic model, and generate synthetic data by sampling from the model. The model is explicit and its parameters are learned through repeated iterations of inference and sampling. The learning objective is applied end-to-end: from training data, to inferred latent representation, and all the way to generated data. **Right**. Empirical Bayes score-based framework: learn a mapping from noisy data to denoised data. The probabilistic model remains implicit and is only revealed by generating samples. There is a single operation, denoising, and it provides a local learning objective: gradually bringing noise closer to the data distribution.

as well as for for sampling. These points are illustrated in Figure 3.2.

Many of the methods reviewed in the previous section (Section 3.1) attempt to learn mappings from training images to Gaussian latent variables, and from latent Gaussian variables to generated images. These mappings are optimized to solve inference and sampling in a single step. What sets the score based approach apart is that it breaks down this process into many independent steps, these steps can be learned separately and are all well constrained. The idea is to train a denoiser to remove arbitrary amounts of Gaussian noise and to use it for sampling: denoising provides local steps between the Gaussian distribution and the data distribution. These local steps can be optimized with a simple least squares procedure, and the trained network contains information about the score function at multiple noise levels. The implicit probabilistic model contained in the network can be revealed via sampling.

Generating samples is reduced to climbing the implicit probability landscape following the direction provided by the denoiser.

Previous work proved that denoisers trained via least squares provide control over the KL divergence between the target distribution and its approximation [111]. Remarkably, it was also shown that common architectural motifs, such as convolution and rectification, give rise to inductive biases that are sufficient to estimate scores from reasonable amounts of data [111]. These methods have been used for solving inverse problems [109] and have fueled a recent burst of progress in generative modeling. These score-based methods are often referred to as diffusion models and have set a new standard in image generation [199, 95, 200]. Corresponding video generation results have been achieved using very large networks and datasets [96, 31].

Problem formulation. Building on this work, we will describe a conditional version of score-based inference. Our goal is to learn and use the distribution of the next frame in image sequences; more precisely, the conditional distribution of the next frame given the past few frames. We will consider a simple architecture that is specialized for temporal prediction and show how to sample probable next-frames. Abstractly, temporal prediction can be thought of an inverse problem that requires prior information in order to recover the part of the information obfuscated by the arrow of time.

We consider a causal non-linear filtering problem [223]: removing undesired noise from an observation in order to recover the underlying signal. The desired distribution of next frames conditioned on the past is $x_{t+1} \sim p(x_{t+1}|c_t)$, where $c_t = [x_t, \dots, x_{t-\tau+1}]$. We assume that the temporal dependencies are local and well approximated by the following conditional independence assumption:

$$\forall t \in \mathbb{N} \text{ and } s \notin \{t, \dots, t-\tau+1\}, \quad x_{t+1} \perp x_s | \{x_s\}_{t \geq s \geq t-\tau+1}. \tag{3.2}$$

We consider architectures with access to τ past conditioning frames, effectively enforcing a finite memory length.

3.2.1 Approximation: learning a family of scores

We aim to learn a conditional denoiser that is both effective at all noise levels, and that automatically estimates the noise level. Such a universal and blind denoiser can adaptively process signals at different distortion level, which we will exploit in the sampling algorithm described in the next section on optimization. We assume that we have access to image sequences and to noisy observations, $y_{t+1} \sim p_\sigma(y_{t+1}|c_t)$, where $y_{t+1} = x_{t+1} + \sigma z$, with $z \sim \mathcal{N}(0, Id)$. We write $g_\sigma(z)$ the probability density function of a Gaussian variable centered at 0 with standard deviation σ evaluated at z. In the remainder of the text, we will drop time subscripts for clarity. We refer to the distribution of the noisy signal given conditioning, $p_\sigma(y|c)$, as the observation density.

The Minimum Mean Squared Error (MMSE) denoising function is given by the posterior expectation:

$$\min_f \mathbb{E}[||x - f(y,c)||^2] \text{ is achieved at } f^*(y,c) = \mathbb{E}[x|y,c]. \tag{3.3}$$

Remarkably this posterior expectation can be expressed as a step in the direction of the score of the observation density scaled by the variance of the noise, an identity often attributed to Stein or to Miyazawa [175, 146]:

$$\mathbb{E}[x|y,c] = y + \sigma^2 \nabla_y \log p_\sigma(y|c). \tag{3.4}$$

The key idea of the score-based framework is to exploit the connection between MMSE denoising and the score function in the reverse direction: first train a denoiser to minimizing MSE, then treat it as an approximation of the scaled score function (which is the optimal solution).

This link between denoising and the score function is at the centre of the score-based framework and it can easily be derived. First express the observation density as a convolution of the target density with a Gaussian distribution of variance σ^2:

$$p_\sigma(y|c) = \int p(x|c) p_\sigma(y|x,c) dx = \int p(x|c) g_\sigma(y-x) dx, \tag{3.5}$$

where the second step uses the conditional independence of y and c given x: $p_\sigma(y|x,c) = p_\sigma(y|x) = g_\sigma(y-x)$. Then differentiate with respect to the noisy signal and divide both sides by the noisy distribution:

$$\begin{aligned} \nabla_y p_\sigma(y|c) &= \int p(x|c) \frac{-1}{\sigma^2}(y-x) g_\sigma(y-x) dx; \\ \frac{\nabla_y p_\sigma(y|c)}{p_\sigma(y|c)} &= \frac{1}{\sigma^2} \int (x-y) \frac{p_\sigma(x,y|c)}{p_\sigma(y|c)} dx. \end{aligned} \tag{3.6}$$

Finally, recognize the score on the left-hand side and rearrange the expression to obtain the desired identity [1]:

$$\sigma^2 \nabla_y \log p_\sigma(y|c) = \int (x-y) p(x|y,c) dx = \mathbb{E}[x|y,c] - y. \tag{3.7}$$

Notice however that the identity in Equation 3.4 requires access to the variance of the noise. Instead we will consider a denoiser that is blind to the distortion level and trained to remove noise of arbitrary magnitude (*i.e.*, universal). Such an optimal denoiser should integrate over noise levels:

$$\hat{x}(y,c) = y + \int \sigma^2 \nabla_y \log p(y|c,\sigma) p(\sigma|y,c) d\sigma. \tag{3.8}$$

In practice estimating the noise level is a simple one dimensional problem that is well con-

[1] This result can also be obtained by first showing the general relationship: $\mathbb{E}_x[\nabla_y \log p(y|x)|y] = \nabla_y \log p(y)$, and then applying it in the Gaussian noise case: $p(y|x) = g_\sigma(y-x)$.

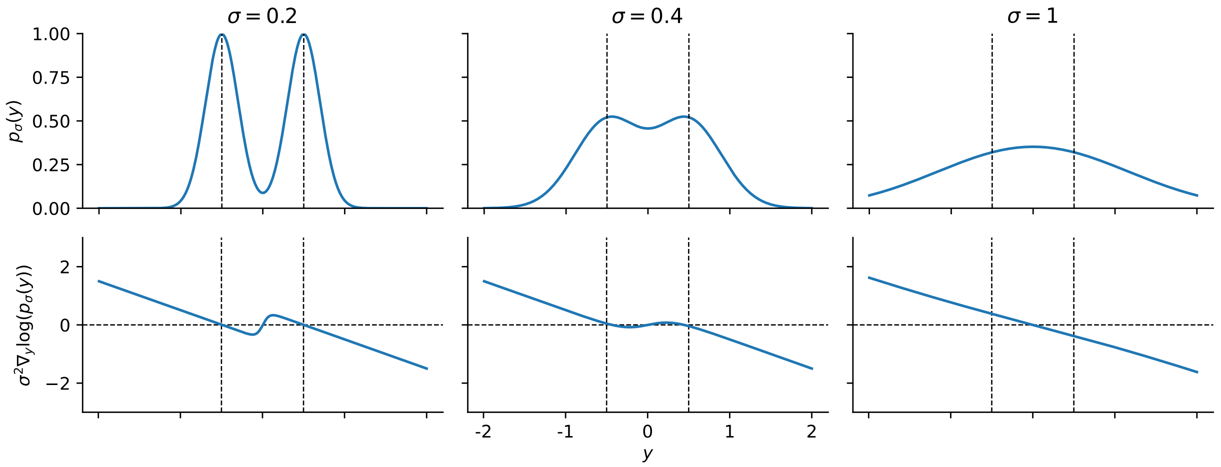


Figure 3.3: Bimodal distribution and score across noise level. One dimensional illustration of the approximation problem. **Top.** Bimodal distribution consisting of two point masses placed at $-1/2$ and $1/2$ (indicated by dashed lines). From left to right, the distribution is convolved with a Gaussian kernel of increasing bandwidth, which corresponds to adding independent Gaussian noise of increasing magnitude. **Bottom.** Optimal denoising step at corresponding noise levels, *i.e.*, the score scaled by the noise variance. The score is the gradient of the log probability with respect to the variable, $\nabla_y \log p_\sigma(y)$. The sampling process described in Section 3.2.2 consists in using these scores to gradually reduce noise (*i.e.*, traverse the family of noisy distributions from right to left on the top row).

strained by the many observed pixels. Note that adding noise to target images is equivalent to smoothing their distribution with a Gaussian distribution, the more noise is added, the larger the bandwidth of the soothing. The continuous family of smoothed distributions forms a scale-space representation of the data distribution [226]. As a result a universal and blind denoiser contains information about that whole family of distributions.

A simple example consisting of a one dimensional distribution comprising two point masses is displayed in Figure 3.3. For this example we consider a pure denoising scenario, there is no conditioning information. At very high noise levels the score points in the direction of the origin. At lower levels of noise, the score points towards the closest of the two point masses of the bimodal distribution. The transition between these two regimes is continuous and smooth.

3.2.2 Optimization: synthesizing new samples

Assuming that we have successfully approximated the family of scores of the observation distribution, we now turn to the question of using the trained denoiser to generate new samples, *i.e.*, predicted next frames conditioned on the past few frames. In the previous section (Eq. 3.4), we showed that the residual of the optimal denoising function, $f(y,c) = \hat{x}(y,c) - y$, is proportional to the gradient of $\log p(y|c)$. The idea is to go along the direction of this gradient and climb towards more probable samples.

Sampling via iterative denoising has demonstrated strong performance on unconditional image generation and can be extended to solve linear inverse problems [109]. Here we describe a procedure for conditional sampling. The algorithm follows the learned scores embedded in a trained denoiser and gradually reduce the effective noise level until a sample is reached. Specifically, starting from an arbitrary image, y_0, iterates a simple map:

$$y_k = y_{k-1} + \alpha_k f(y_{k-1}, c) + \gamma_k z_k, \tag{3.9}$$

where the parameter α_k controls the step-size and the parameter γ_k controls the amplitude of a sample of additive noise. It is natural to pick a geometric schedule for the step-size. Adding noise along the sampling iterations, $z_k \sim \mathcal{N}(0, Id)$, avoids getting stuck in local maxima and promotes exploration. An additional parameter $\beta \in [0,1]$ controls the proportion of injected noise, playing the role of an inverse temperature. The key idea of this procedure is to chose the amplitude γ_k so as to reduce the effective noise magnitude on each iteration[2].

Initially the algorithm takes larger steps, which correspond to removing a lot of noise, and gradually makes smaller adjustments as the generated image starts to look more like a plausible sample. This coarse-to-fine iterative refinement behavior is central to the success

[2] The amplitude of the additive noise is set to: $\gamma_k^2 = ((1-\beta\alpha_k)^2 - (1-\alpha_k)^2)\sigma_k^2$ where the effective noise level is $\sigma_k = ||f(y_{k-1}|c)||/\sqrt{d}$, see [109] for the derivation.

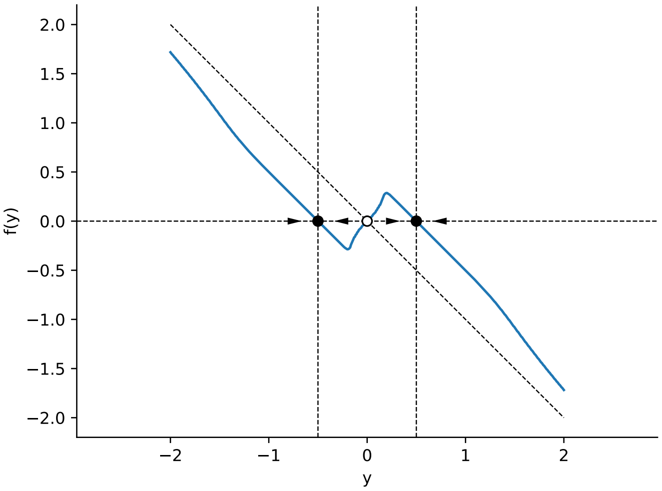


Figure 3.4: Blind universal denoiser for bimodal distribution. One dimensional illustration of the optimization problem. Residual of the optimal denoising function for the distribution displayed in Figure 3.3. The function was computed numerically with a plug-in estimator for the standard deviation of the noise. There are two stable fixed points on the point masses of the target distribution and an unstable fixed point between the two. The sampling algorithm will reach one of these fixed points depending on initial conditions and the effect of added noise.

of the sampling algorithm. Indeed local search methods can only explore high-dimensional landscapes that are sufficiently smooth (log-concave). The universal-blind denoiser effectively relies on such a blurred distribution, which enables sampling.

This sampling algorithm is illustrated in the simple one dimensional bimodal distribution in Figure 3.3. We numerically evaluated the optimal denoiser (recall that in this simple case there is no conditioning). For simplicity we used a maximum a posteriori (MAP) plug-in estimator for the effective noise level: $\hat{\sigma} = \text{argmax}_\sigma p(\sigma|y)$. We considered a non-informative prior for the noise level: $p(\sigma) \propto 1/\sigma$, *i.e.*, a log-uniform distribution. The optimal denoiser residual is given by:

$$f(y) \approx \hat{\sigma}(y)^2 \nabla_y \log p(y|\hat{\sigma}(y)), \tag{3.10}$$

and is illustrated in Figure 3.4. The sampling algorithm will chose between the two point masses depending on the initial condition and the effect of noise added during sampling.

3.3 Numerical results

Architecture and conditioning mechanism. In this section we will use a U-net [176] composed of three scales. Past conditioning frames are concatenated as extra input to the network. This conditioning mechanism was also used to denoise fine scales wavelet coefficients based on coarse scales coefficients [110]. Other common conditioning approaches include adjusting the gain and bias at each layer (*i.e.*, two numbers per channel, delocalized), and cross attention mechanism (query form conditioning, key and value from noisy observation). Concatenating past frames as additional input has the advantage of being elementary and it is also well suited to the temporal prediction case. Indeed, conditioning frames have the same shape as the noisy observation, and local convolutional processing can make use of these two sources of evidence for the denoising task. The number of conditioning frames can be varied by modifying the number of input channels (denoted τ). More details on the architecture are provided in Appendix 5.

3.3.1 Tackling occlusions on a procedural dataset

Occlusion boundaries are an inevitable result of image formation and play an important role in depth perception. Traditional motion compensated technique fail to account for the nonrigid motion patters near occlusion boundaries.

Moving leaves. We designed a synthetic dataset in order to reduce the occlusion problem to a minimal case. Our aim is to use this dataset to demonstrate how the score-based inference framework defined in Section 3.2 makes decisions on ambiguous occlusion sequences.

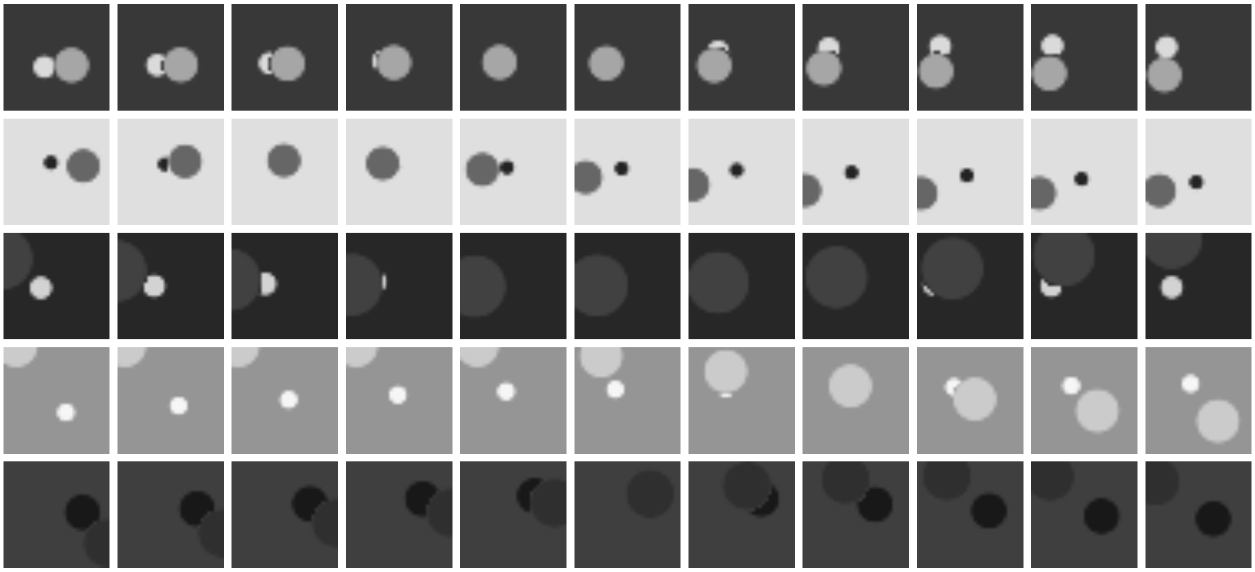


Figure 3.5: Moving leaves dataset. Five example image sequences, each containing two disks against a blank background. Disks move along smooth curves and occlude each other, with the larger disk always occluding the smaller one.

Building on the dead leaves model defined in the natural scene statistics literature [142, 129, 14], we propose a moving disks datasets. Disks are randomly dropped on an image canvas and occlude each other, we also assign a random depth (or distance) to each disk. The disks are assumed to have the same physical size and to stay at their fixed depth. Each disk is then moved randomly along smooth trajectories to generate image sequences that respect occlusion relationships. Within each image sequence, the disks do not change in their projected image size. The trajectories of each disk are sampled from a Gaussian processes, and their average speed of motion scales inversely with depth (with objects at a further distance moving more slowly). These image sequences contain two reliable, albeit indirect, cues to depth: the projected disk size, and motion parallax. The luminance of each disk and of the background are selected uniformly at random between 0 and 1. More details on this dataset are provided in Appendix 5.

There are several benefits to using such a procedural dataset: it enables sampling a distribution along continuous features such as size, speed, texture, *etc.*; and it is easy to change the rules and control the difficulty of the task. For example allowing the disks to move along the depth axis, thereby decoupling size from range; or changing the occlusion rule, letting brighter disk occlude for example, or making the occlusion rule stochastic; or changing the number of disks, *etc.* This dataset is also useful to study the representation of fast small object, which are subject to temporal aliasing.

Results. A U-net with two conditioning frames ($\tau = 2$) was trained on the moving leaves dataset. The network was evaluated on new sequences of disks which were not included in the training set. This probe sequence contains two clean conditioning frames and the target next frame is initialized with pure noise. Samples of probable next frame can be generated using the sampling algorithm described in Section 3.2.2. Example samples around an unambiguous occlusion boundary are displayed in Figure 3.6. Similarly, example samples around an ambiguous occlusion boundary are displayed in Figure 3.7. The network can be used to sample occluded disks which respect the dataset property whereby smaller disks get occluded. Notice that the occluder disk tends to be almost perfectly round while the occluded one is often deformed. These results highlight the power of the probabilistic sampling algorithm as compared to deterministic one step prediction.

The degree of ambiguity of a disk occlusion was then varied continuously by controlling the relative size of two disks moving towards one another. The results are summarized in Figure 3.8. Notice that the network adapts its predictions to the relative size difference between the two disks, sampling exclusively one kind of occlusion when the measurement are unambiguous and gradually becoming more stochastic when they are ambiguous. We note that these effects are only observed in well trained networks and small gains in denoising performance can have a large impact on the quality of generated samples.

Finally, we recursively generated longer sequences to evaluate ability to generate temporally coherent sequence and revisit the example from Figure 3.1. Two such sequences are

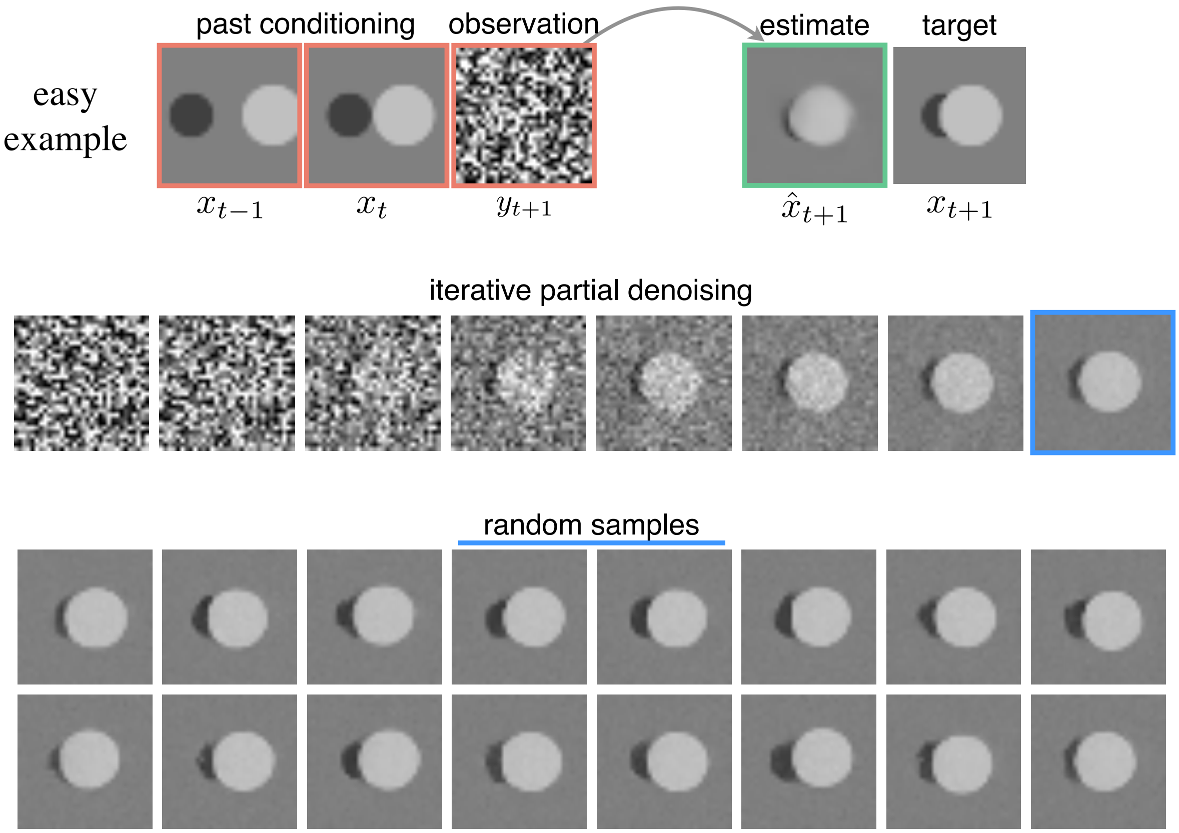


Figure 3.6: Samples around unambiguous occlusion boundary. Predicting an unambiguous disk occlusion. **Top.** Two conditioning frames contain disks of different size moving towards each other. The network observes pure noise in the next frame and estimates the target (all input frames are highlighted in red). In this example, the next frame is unambiguous: the light disk on the right should occlude the other. As expected, the denoising estimate (highlighted in green) is a close approximation of the target (but the edges of the disk are blurry). **Middle.** Intermediate steps of the iterative partial denoising procedure, this score-based sampling algorithm uses the same conditional denoiser network as above. The corresponding sampled probable next-frame is highlighted in blue. **Bottom.** Example samples of probable next-frame generated using the iterative partial denoising procedure starting from different random initializations. Each contain sharp occlusion boundaries, all with the light disk on the right occluding the other. With this sampling procedure, the network decides on the occlusion and produces diverse samples that, unlike one step denoising, do not blur the edges.

displayed in Figure 3.9. The network was initialized with two images from the test set and successive frames were sampled recursively. The sampled image sequences are coherent over many frames and yet different from the sequence in the test set. Notice that the network

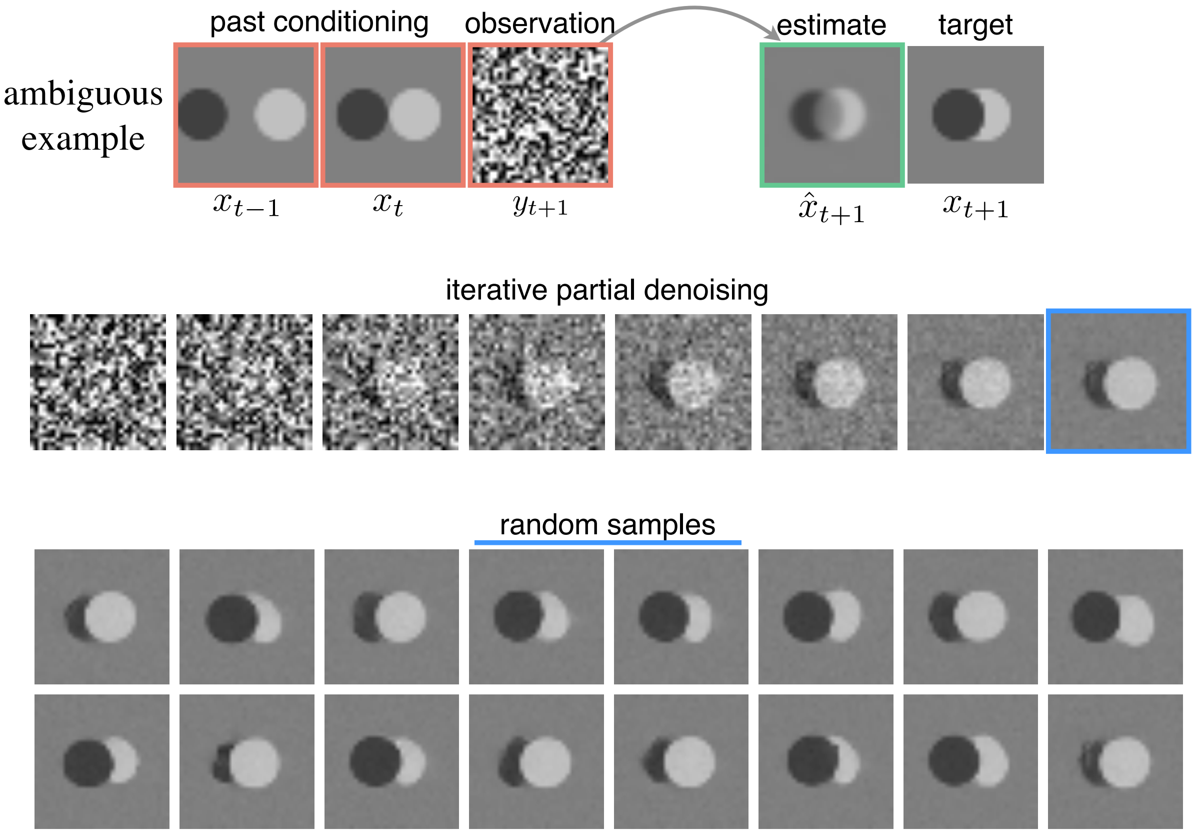


Figure 3.7: Samples around ambiguous occlusion boundary. Predicting an ambiguous disk occlusion. **Top.** Two conditioning frames contain disks of equal size moving towards each other. The network observes pure noise in the next frame and estimates the target (all input frames are highlighted in red). In this example, the next frame is ambiguous: there are no cue as to which disk should occlude the other. As expected, the denoising estimate (highlighted in green) is a blurry mixture between the two possibilities (it is an average over the whole posterior). **Middle.** Intermediate steps of the iterative partial denoising procedure, this score-based sampling algorithm uses the same conditional denoiser network as above. The corresponding sampled probable next-frame is highlighted in blue. **Bottom.** Example samples of probable next-frame generated using the iterative partial denoising procedure starting from different random initializations. Each contain sharp occlusion boundaries, with either disk occluding the other at about equal frequency. With this sampling procedure, the network decides on the occlusion and produces diverse samples that, unlike one step denoising, do not compromise between the two possibilities.

tends to drop or modify disks that are occluded in the first example and creates a second disk out of nothing in the second example. Moreover, we see the sampling algorithm starting to fail at on the last frame of the second example. In other examples we have observed that

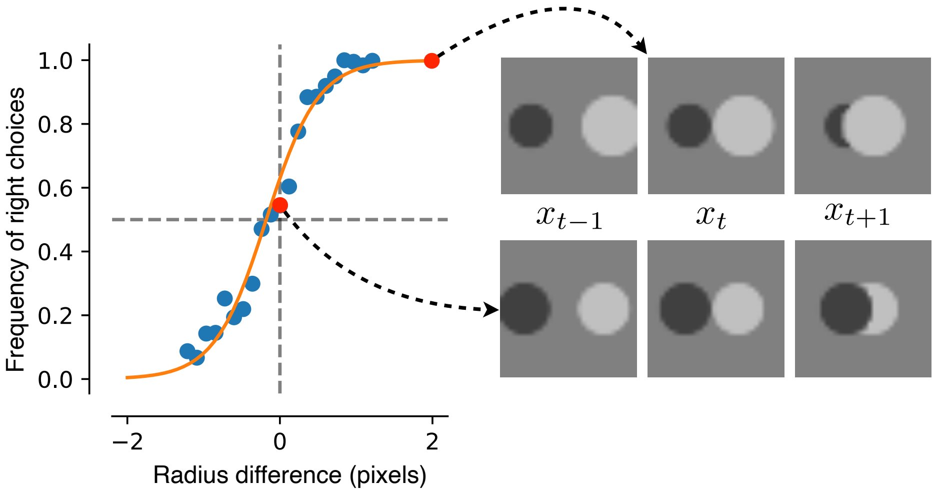


Figure 3.8: Decisions around occlusion boundaries. Showing frequency of right disk occlusion as a function of the difference in depth between the two disks (averaged over 64 samples). Highlighted in red are the unambiguous examples from Figure 3.6 and the ambigous example from Figure 3.7. A psychometric function is fit to the choice data and shows that the network is sensitive to the disk's relative size difference (and is slightly biased).

small disks often do not reappear after being fully occluded. This indicates a need for longer temporal dependencies, the restriction to two conditioning frames being too strict, especially in presence of acceleration.

3.3.2 Probabilistic prediction of natural videos

Denoising performance. Several U-nets with varying conditioning lengths were trained for next frame prediction on a generic dataset of natural image sequences (see description in Appendix 5). The test performance of trained denoiser networks are summarized in Figure 3.10. In the unconditional case (*i.e.*, $\tau = 0$), the network only exploits static image structure and recovers the image denoising performance described in [147]: the error curve meets the identity line at 40 dB and has a slope of approximately one-half. For conditional image denoising, there is a gradual performance improvement that is particularly saliant at low input PSNR values. The largest increases in output PSNR is obtained from conditioning

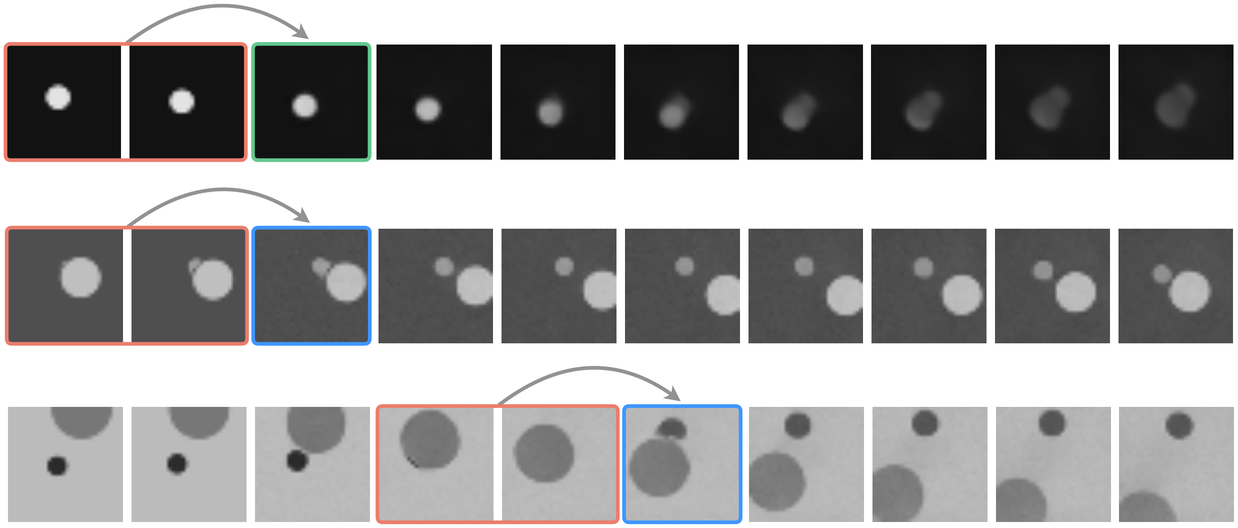


Figure 3.9: Generated moving disks sequences. Coherent sequences can be generated using a conditional denoiser via recursive sampling, but not via recursive estimation. In all three examples the first two frames come from the test dataset and the successive frames are generated recursively, using the previous two frames as conditioning. **Top.** Recursive estimation obtained by estimated a denoised next frame in one step. For the first step of this recursive process, the inputs to the conditional denoiser are highlighted in red (the noise observation, y_{t+1}, is not shown) and the estimated next frame is highlighted in green. This estimation is blurry because it approximates the posterior mean. After a few recursive steps predictions collapse to an uninformative blurry image, analogous to the situation in figure 3.1. **Middle.** Recursive sampling of probable next frame obtained via iterative partial denoising using the same denoiser as above. The generated next frame (highlighted in blue) is sampled from the conditional distribution of probable next frame. Recursive samples generate a coherent image sequence with disk motion and occlusions (not shown in this particular example). **Bottom.** Another example of recursive sampling. The forth step in the recursive sampling process is highlighted and shows that after full occlusion of the small disk, its size, color and location is ambiguous. This example shows the limitations of the plain conditional denoising approach due its short memory length—only accessing two past conditioning frames.

on one baseline image (*i.e.*, $\tau = 1$). Adding a second conditioning frame (*i.e.*, $\tau = 2$) improves performance further, which is indirect evidence that the network is making use of motion information. The benefits of conditioning reach a plateau with three past frames. The trained denoiser networks with two and three conditioning frames also match pure prediction performance which is indicated by a vertical dashed line at 28dB. Indeed at

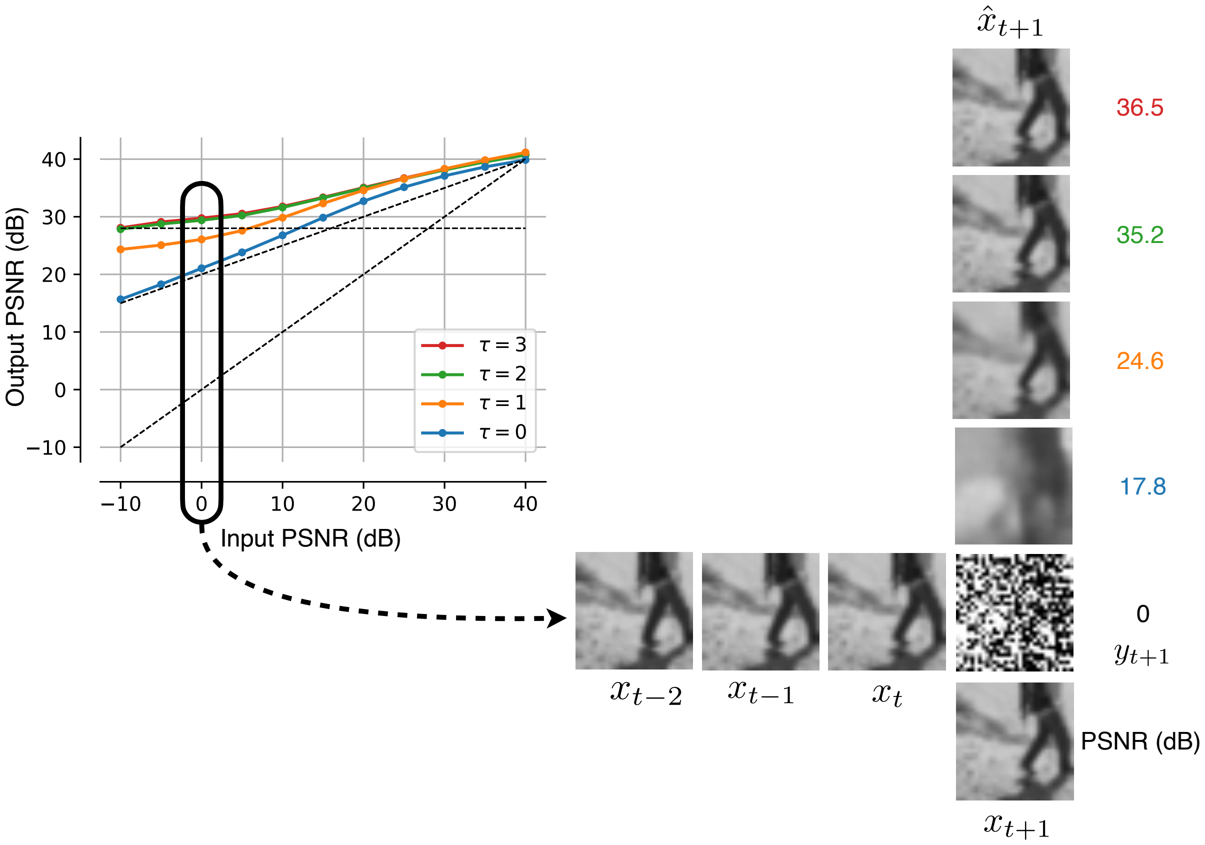


Figure 3.10: Conditional image denoising performance. Denoisers with varying number of conditioning frames (τ) exploit spatio-temporal structure of natural image sequences. **Left.** Input-output PSNR curves summarize the test performance of trained denoisers. The horizontal axis represents difficulty, with lower PSNR value corresponding to stronger distortion. The vertical axis represents performance, with higher PSNR value corresponding to better denoising. Dashed black lines indicate the identity, slope one-half, and a horizontal line at 28dB corresponding to the performance of the same network trained for prediction alone (two conditioning frames, no noisy observation). **Right.** Horizontally: example image sequence, noisy observation and target next frame at 0dB input PSNR. Vertically: estimated next frame for denoisers with varying number of conditioning frames (with matching color code). Longer memory results in higher quality denoising, showing that the denoiser can exploit spatio-temporal image structure.

very low input PSNR values there is effectively no information in the observation and the network's performance is entirely driven by temporal prediction. Estimated next frames for denoisers with varying number of conditioning frames are shown for an example image sequences in Figure 3.10.

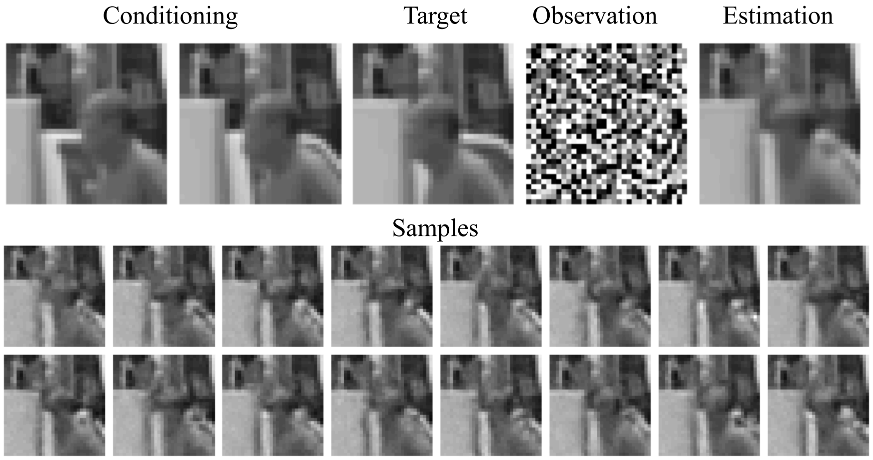


Figure 3.11: Samples of probable next-frame. Top. Example image sequence from the DAVIS test set. The two conditioning frames show a man walking to the left and a car driving to the right in the background. In natural image sequence, the person occludes the car in the next frame. Starting from an observation of pure noise the network estimates a blurry next frame. **Bottom.** Samples starting from different noisy initializations are shown below. They are diverse but their perceptual quality is not very high. In particular the features of the face are lost.

Samples of probable next-frame. We then sampled from the trained denoiser with two conditioning frames. The results are shown in Figure 3.11 but they have low perceptual quality. Sampling diverse and high quality temporally stable trajectories along the manifold of natural images requires a larger training set and more training iterations. It may be possible to improve the quality of next-frame samples by considering architectures with richer non-linearities, as was described in Chapter 2.

Spatio-temporal adaptive linear filtering. The U-net architecture used here is bias-free, *i.e.*, there are no additive constants in the convolutional layers or the batch-norm layers. Such a network is homogeneous and locally linear and as a result can be interpreted as computing an adaptive linear filter [147]. Specifically, the conditional denoiser can be expressed as a linear function of the conditioning frame and the noisy observation:

$$\hat{x}(y,c) = \hat{x}_y + \hat{x}_c, \text{where } \hat{x}_y = \nabla_y \hat{x}(y,c) \cdot y, \text{ and } \hat{x}_c = \nabla_c \hat{x}(y,c) \cdot c. \tag{3.11}$$

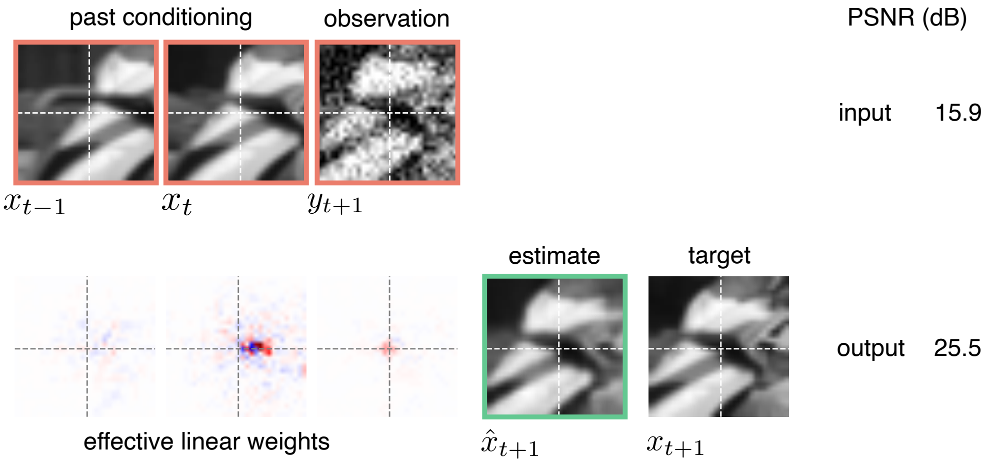


Figure 3.12: Spatio-temporal adaptive filtering. Visualization of the effective linear weights used to compute the central denoised pixel of an example image sequence. **Top.** Image sequence from the test set, a pattern is moving to the left. Dashed white lines highlight the central pixel of each frame for reference. The two past conditioning frames are clean and the noisy observed frame has a PSNR of about 16 dB. **Bottom.** Effective linear filter used by the network to weight the conditioning frames and the noisy observation, y_{t+1}, to estimate the center pixel of the next frame $\hat{x}_{t+1}$. Each pixel of the estimation is effectively computed by weighting the inputs with some linear filter (only the one corresponding to the center pixel is shown here). Notice that the effective filter locally averages the noisy observation and focuses on the part of the previous image displaced to the right (corresponding to leftward motion of the pattern). The estimated frame has a much higher PSNR of about 25.5 dB.

We evaluated those Jacobian matrices on the trained denoiser and display the result in Figure 3.12. The linear filters are oriented in space-time and adaptively track the motion of the underlying pattern. This is analogous to the observations made on video denoising networks [189].

Adaptively weighing evidence by reliability. Since the network utilizes both past conditioning frame and noisy future observation, we can ask how it combines these two sources of information to produce an estimated next frame. In general, the decomposition of the posterior over two cues is:

$$p(x|y,c) = \frac{p(x,y,c)}{p(y,c)} = \frac{p(y|x)p(c|x)p(x)}{p(y,c)} = \frac{p(x|y)p(x|c)}{p(x)} \frac{p(y)p(c)}{p(y,c)}, \tag{3.12}$$

where we use conditional independence in the second step. The two individual posteriors are combined multiplicatively with a third interaction term that quantifies how independent the two cues are. We can write an analogous decomposition of the squared denoising error:

$$||x - \hat{x}(y, c)||^2 = ||x - \hat{x}_c||^2 + ||x - \hat{x}_y||^2 - ||x||^2 + 2\langle \hat{x}_y, \hat{x}_c \rangle, \tag{3.13}$$

where we reuse the notation for locally linear effective computation from Figure 3.12. This expression is a partition of variance and it reveals how each cue contributes to the overall denoising performance. We evaluated the network performance on three probe sequences at different noise levels. We also computed a local linear approximation of the network at each of these levels and computed the first two terms on the right hand side of equation 3.13. The results are displayed in Figure 3.13 shows that the network appropriately combines evidence weighing it by its reliability. Note that the reliability of past frames and of noisy observation are estimated automatically, *i.e.*, the network performs blind evidence integration.

This result shows that conditioning has more impact on the estimate at high noise level. As the noise level is reduced, the model relies gradually more on the observation and finally ignores the conditioning altogether. This means that at some point in the sampling process, the network effectively acts as a image generation model. Such an image generation model is no longer constrained by past frames and will explore the image manifold locally. This may help explain the poor sample quality that was observed in Figure 3.11. It would be interesting to develop a denoising architecture that more explicitly controls the role of past conditioning. Intuitively, the geometrical picture corresponding to denoising and prediction are complementary. Denoising corresponds to projecting a distorted image on the image manifold, while prediction amounts to extrapolating along the manifold.

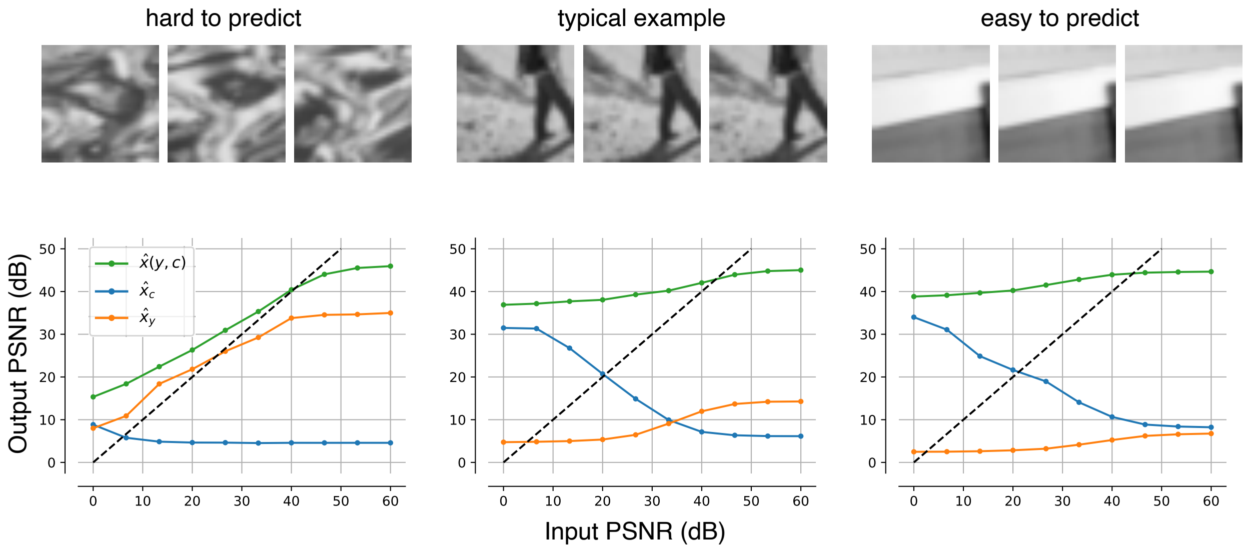


Figure 3.13: Adaptive cue combination. Local linear analysis of information integration from two different sources, past conditioning and noisy observation. In green the denoising performance for a single probe image sequence as in Figure 3.10. In orange the denoising performance due to noisy observation only, it increases with input PSNR. In blue the denoising performance due to past conditioning frames only, it decreases with input PSNR. These two curves are calculated using a local linear approximation as described in Eq. 3.11. The level at which the orange and blue curve cross depends on how difficult the prediction is, demonstrating that the network adaptively weights evidence by reliability. **Top.** A texture image sequence with non rigid complex motion. The orange and blue curves cross around 0dB input PSNR. **Middle.** A person walking to the right on a stable background. The orange and blue curves cross around 30dB input PSNR. **Bottom.** A static pattern consisting of three smooth regions separated with straight boundaries. The orange and blue curves cross around 60dB input PSNR.

3.4 Discussion

In this chapter we cast probabilistic temporal prediction as a generative modeling task and describe an empirical Bayes method to generate probable video predictions. The approach is motivated by the incomplete and noisy nature of visual measurements and the conditional nature of inference through time. We showed how to embed the distribution of future signals given past observations in a network via conditional denoising, and we put this model to use by generating probable next-frames. The sampled next-frames correctly handle ambiguous situations, choosing a depth order for objects that occlude each other. This ability to handle multi-modal distribution is critical and stands in contrast to methods that blur the uncertain regions, especially along occlusion boundaries. The moving leaves procedural dataset was designed to illustrate the fact that, even in elementary mechanical scenarios, probabilistic modeling is appropriate. Indeed, measurements are always incomplete and the probabilistic framework can readily handle this ambiguity. Moreover the network displays hallmarks of probabilistic computation: it adaptively combines information from past conditioning and noisy observations, appropriately weighting them according to their reliability.

The simplicity of the score-based empirical Bayes framework is appealing and its success suggests that density estimation and sampling—although very challenging in high-dimensions—may be tackled with well known and robust tools: (non-linear) least squares regression and (coarse-to-fine) gradient ascent. This is only possible because i) noise offers a local learning signal and enables sampling, and ii) the network architecture matches the structure of the visual signal being processed.

What is noise good for? The denoising formulation of empirical Bayes framework reveals that noise can play an important functional role in both estimation and sampling. This functional role can be summarized by three properties: regularity, locality, and uniformity. First, noise enables sampling because it regularizes the learned energy landscape, making it easier to learn and to climb. Indeed, adding noise smooths the distribution (making it more log-concave) and thereby amenable to local search methods (both learning and sampling are gradient based). Adding noise during sampling also drives exploration, allowing the samples to escape from local maxima, and promoting diversity. Second, resilience to noise is a local objective in the sense that it brings data from random Gaussian to the target distribution incrementally. Such locality in the space of densities offers a powerful simplification: the transformation from data to Gaussian and back to data is broken down into steps. As a result, denoising provides a learning signal for each of these intermediate steps and allows learning to proceed in parallel. This locality may explain why the denoising objective is more efficient than traditional end-to-end methods. Third, high-dimensional random normal variables concentrate on a hypersphere whose radius is the square root of the dimensionality and tile its surface uniformly. This uniformity of random Gaussian samples in high dimensions is desirable for calculating expectations, which is central to both inference and sampling. Indeed, such an isotropic distribution (*i.e.*, rotationally invariant) provides an efficient grid on which to evaluate high-dimensional integrals. In contrast, the distribution of points on a regular grid is far uniform in high dimensions (*i.e.*, not rotationally invariant), and the question of designing uniform grids in high dimensions with deterministic methods remains open.

Implicit bias of the architecture. Although the trained deep network achieves accurate conditional denoising performance, it remains difficult to understand the functional logic of its computation. The local linear analysis starts to reveal some of this prediction mechanism. The internal computation could be further exposed by an eigen-decomposition of the Jacobian. Of particular interest is the possibility to relate these eigen-values and eigen-vectors back to the polar prediction mechanism of Chapter 2. Turning the polar prediction model into an analysis tool may help explain the origin of the previously reported "geometry adaptive harmonic bases", and elucidate whether they come in pairs [111]. In turn, such an analysis can motivate architectural modifications, perhaps involving richer non-linearities such as local gain-control. In this chapter we only considered standard U-nets, and conditioning information was provided as an additional input, via concatenation. A more explicit conditioning mechanism may change the implicit bias of the architecture. Finally, there is also an opportunity to reformulate the sequence-to-image architecture into an encoder-decoder like architecture, and to extract latent features—leveraging temporal prediction to learn abstract representations that are informed by the statistics of image sequences.

Chapter 4

Modeling biological vision

> "*The principle that the redundancy in sensory messages resulting from regularities in the environment are exploited in sensory pathways illuminates a host of sensory phenomena, [. . .], and possibly the nature of intelligence itself.*"
>
> - Horace Barlow (2001), *The exploitation of regularities in the environment by the brain*

The modern history of neuroscience is punctuated by the discovery of neurons along the visual hierarchy that respond to features of the environment: from center-surround filters in the retina and the visual thalamus [126, 62], to the characterization of orientation and spatial frequency selectivity in the primary visual cortex [101], and up the dorsal and ventral streams with motion [148] and texture [91] selectivity respectively. The slow but steady progress in describing how neural activity relates to perception has fueled a vast field of research and has positioned vision as a model system of choice to study a broad range of topics from biological mechanism to cognitive function.

Canonical computational elements. Neural selectivities can be described as being computed from simple canonical building blocks [56]. Elements like linear filtering, rectifying non-linearity, local pooling and gain control have proven remarkably effective at capturing the activity of neurons in the early primate visual system. When combined into cascaded model, canonical computational elements have proven to be very expressive and powerful. This modular design has the advantage of relying on elements that are relatively simple and can be understood precisely. Although such models do not reveal the exact circuit basis of visual perception, they can provide a functional account of visual processing. Moreover, these elements seem to be appropriate for describing local circuits in many cortical areas [36]. This has given rise to the hypothesis that cells in each cortical area perform the same form of computation—an idea that is foundational to deep neural networks [74, 173].

Yet, mid and high-level visual areas remain difficult to understand. Even for the subregions where selectivity of neurons has been described (*e.g.*, the face patches in IT [94]), we still do not know how selectivity is constructed from the signals afferent to these neurons. There are at least three main reasons why characterizing the selectivity of visual neurons for environmental features is challenging: i) the space of natural images is too large to be explored exhaustively, ii) the quest for effective canonical computational elements is difficult to constrain, and iii) the visual system continually adapts as it is being probed by experimenters seeking which stimulus to show. As a result, there is currently no computational account for the selectivity of neurons and populations in these regions, *i.e.*, we do not have "image-computable" models of those regions.

Deep network models of visual processing. The advent of deep neural networks has spurred a new wave of visual processing models that are based on large natural visual data and tasks. These networks correlate with recorded neural firing rates from the primate mid- to high-level visual cortex and outperform previous approaches [231, 186]. These methods have been tested using a controlled experimental paradigm where static images are flashed in random order and neural responses from macaques are recorded and averaged over repetitions of the same image.

Although these computational models are fully accessible to analysis, it remains challenging to understand the functional properties that underlie their successes and limitations. In particular, their neural predictivity is sensitive to choices of objective function, architecture, dataset and training methodology—which limits our ability compare networks to biological circuits or perceptual behavior. Moreover, using neural recordings as another benchmark for deep networks risks over-fitting to a particular set of physiological measurements while it remains unclear which measurements are most worth predicting (ion channel dynamics, blood oxygenation level, or else, and in response to artificial stimuli, or natural movies, etc.).

Despite their limitations, deep networks have boosted the top-down behavioral approach to modeling vision and demonstrated the feasibility and effectiveness of feedforward models composed of simple modules. In this chapter, we will also adopt a performance optimization approach to modeling biological vision. We extend this approach beyond object recognition and consider the more general temporal prediction objective (which has the advantage of being unsupervised). We also extend the architecture beyond convolution-rectification and consider biologically motivated and powerful canonical computations such as local gain control and quadratic non-linearities. Our goal is to capture the functional properties and responses of neurons to image sequences, including their dynamics.

Representing transformation groups. In their seminal work on modeling visual computation, Pitts and McCulloch adopted a geometrical framework to describe perceptual invariance [159]. They described a method for invariant object recognition that averages along group orbits, and they also proposed a mechanism for aligning objects to a standard form. The emphasis was on explaining how object categorization can ignore nuisance variability in position, orientation, distance, luminance, etc. More generally, the idea that architecture and properties of neurons reflect the symmetry properties of the physical world has motivated many models of visual processing [4, 160].

We also consider learning to represent transformation groups, and aim to go beyond invariant object recognition. We focus on factorizing form and pose for temporal prediction, emphasizing the importance of equivariance to physical transformations. Equivariance is a property of representations which are sensitive to signal variability. Specifically, a representation is equivariant when there is a correspondence between transformations of the signal and transformations of the representation (this idea is illustrated with a commutative diagram in section 2.6). Indeed variability in pose is not nuisance, instead it is actively represented in the ventral stream where perceptual and neural sensitivity have been demonstrated [97]. Our hypothesis is that predicting transformation dynamics drives a separation between a slow invariant component (the “what”) and a linearized equivariant component (the “where”). Specifically, small population of neurons should use this transformation structure to locally predict their upcoming inputs. In other words, we posit that neurons should extract and exploit the signal structures that can be modeled as transformation groups.

In this chapter, we discuss the relevance of the polar prediction model to understanding the nonlinear response properties of neurons in the primate visual system. Our goal is to derive a model for sensory computation from the principles of visual temporal prediction described in Chapter 2. First, we show how the computations of the polar prediction model can be mapped to known physiological elements of the early visual system. We then extend this approach to a second layer and treat polar prediction as a module that can be cascaded. We end this chapter with a discussion of some perceptual and physiological evidence, and highlight important open problems in modeling visual temporal prediction.

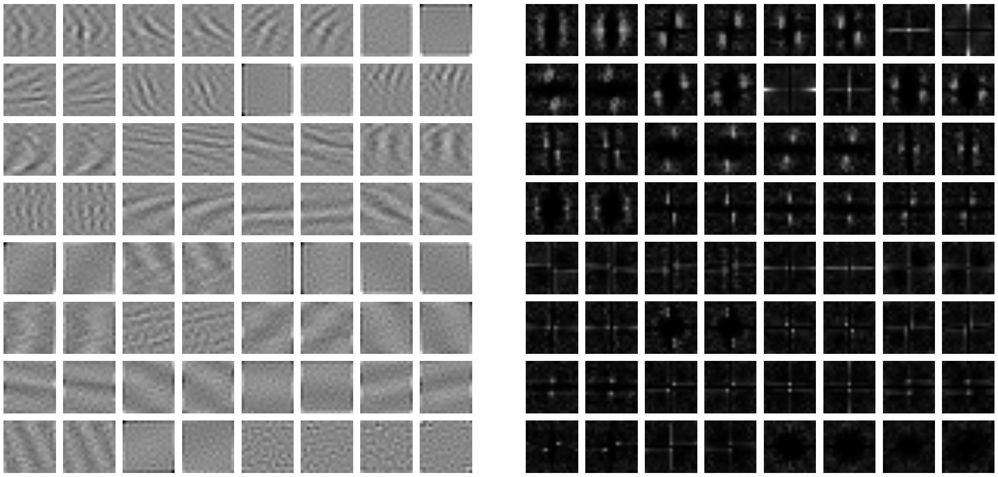


Figure 4.1: Learned polar predictor filters. Example filters from a Polar Predictor model trained to predict videos from the DAVIS dataset. **Left.** Spatial domain filters of size 17×17 pixels. All 32 quadrature pairs of filters are displayed side by side and sorted by their norm. **Right.** Corresponding Fourier amplitude spectra displayed similarly. Observe that the filters are selective for orientation and spatial frequency.

4.1 Neural circuit computation of polar prediction

The polar prediction model described in Chapter 2 uses a phase extrapolation mechanism that builds-in a polar non-linearity and automatically learns quadrature pairs of filters. A polar predictor composed of 32 pairs of convolutional filters was trained on the DAVIS dataset, and the learned filters are displayed in Figure 4.1. These filters resemble receptive fields of neurons in area V1 (primary visual cortex). They are selective for orientation and spatial frequency, filters in each pair have similar frequency selectivity. But a lot of these structures were built into the architecture as opposed to learned from data. Previous models also imposed a polar change of coordinates to model neural computation of visual representations [35] (see discussion in section 2.4). In this section we relax these assumptions, learn the prediction mechanism jointly with the filters, and bring the model closer to biological neural circuits.

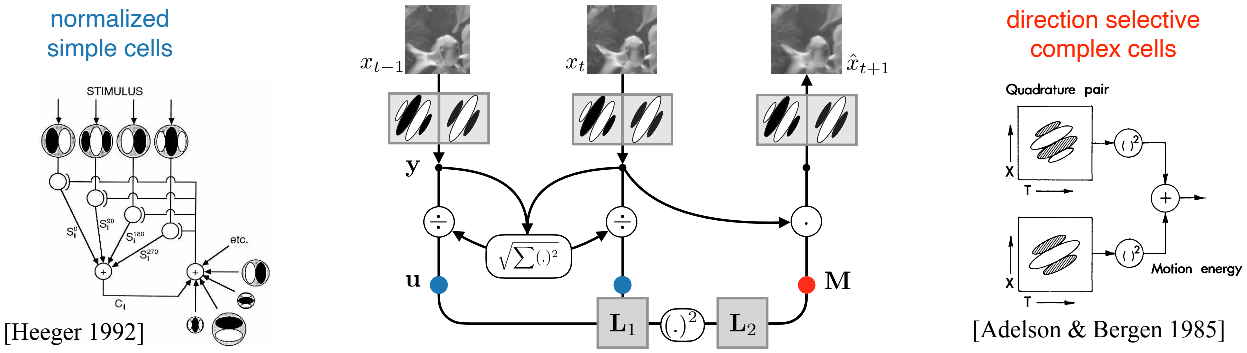
4.1.1 Quadratic predictor

Polar prediction can be restructured into components resembling normalized simple and direction-selective complex cell models of primate V1 neurons. This correspondence goes beyond the qualitative match of learned filters from Figure 4.1 and suggests a plausible circuit implementation for temporal prediction. Building models from elements that are consistent with know physiology offers a path to explaining the non-linear response properties of cells in primate visual cortex.

Circuit computation. Let us derive a circuit algorithm for temporal prediction based on canonical computational elements. We consider a parameterized predictor mechanism (P_w) that can be learned jointly with the analysis and synthesis filters. This "quadratic predictor" (hereafter **QP**) generalizes the polar extrapolation mechanism and can accommodate groups of channels of size larger than two (polar predictor filters only have a real and an imaginary part). As such, the quadratic predictor subsumes phase extrapolation as a spatial case (and recovers the polar non-linearity on synthetic data).

The quadratic prediction mechanism is depicted and described in Figure 4.2. In the case of pairs of filters, $g = 2$, the quadratic prediction mechanism can be described as follows. Current and previous pairs of coefficients ($\mathbf{y}_{k,t} = [y_{2k,t}, y_{2k+1,t}, y_{2k,t-1}, y_{2k+1,t-1}]^\top \in \mathbb{R}^4$) are normalized ($\mathbf{u}_{k,t} \in \mathbb{R}^4$), then linearly combined ($\mathbf{L_1} \in \mathbb{R}^{4\times d}$), squared (elementwise), and linearly combined again ($\mathbf{L}_2 \in \mathbb{R}^{4\times d}$) to produce a prediction matrix ($\mathbf{M}_{k,t} \in \mathbb{R}^{2\times 2}$) that is applied to the current coefficients to produce a prediction ($[\hat{y}_{2k,t+1}, \hat{y}_{2k+1,t+1}]^\top \in \mathbb{R}^2$). These linear combination can be learned jointly with the analysis and synthesis weights by minimizing the prediction error.

This quadratic prediction mechanism, still in the case of pairs of filters, can be derived from the phase extrapolation mechanism, by writing it using only real-valued elements. First express the filter responses in polar coordinates: $y_{2k,t} = \mathbf{v}_{2k}^\top \mathbf{x}_t = a_{k,t}\cos(\theta_{k,t})$, and



[Heeger 1992]

[Adelson & Bergen 1985]

Figure 4.2: Quadratic prediction mechanism. Computing temporal predictions with a quadratic prediction circuit that unifies two classical neural models. **Left.** The divisive normalization model describes gain control in V1 simple cells and accounts for the effects of contrasts and masking [89]. Cell responses are divided by the pooled activity of neighboring units. **Right.** The spatio-temporal energy model for the perception of motion [3]. Two direction selective simple cells in quadrature are squared and combined linearly to extract a motion energy signal. **Center.** Groups of coefficients ($\mathbf{y}$) at the previous and current time-step are normalized ($\mathbf{u}$) and then passed through in a Linear-Square-Linear cascade to produce a prediction matrix ($\mathbf{M}$). This matrix is applied to the current vector of coefficients to predict the next one. The filters as well as the linear transforms ($\mathbf{L}_1$ and $\mathbf{L}_2$) are learned. This quadratic prediction module contains phase extrapolation as a special case and handles the groups of coefficients of arbitrary size (*i.e.*, going beyond pairs). Blue dots indicate units that resemble normalized simple cells, and the red dot indicates units that resemble direction selective complex cells.

$y_{2k+1,t} = \mathbf{v}_{2k+1}^\top \mathbf{x}_t = a_{k,t} \sin(\theta_{k,t})$. Then write phase advance as an explicit two dimensional rotation:

$$\begin{bmatrix} \hat{y}_{2k,t+1} \\ \hat{y}_{2k+1,t+1} \end{bmatrix} = \begin{bmatrix} \cos \Delta\theta_{k,t} & -\sin \Delta\theta_{k,t} \\ \sin \Delta\theta_{k,t} & \cos \Delta\theta_{k,t} \end{bmatrix} \begin{bmatrix} y_{2k,t} \\ y_{2k+1,t} \end{bmatrix}, \tag{4.1}$$

where the 2×2 prediction matrix, $\mathbf{M}_{k,t}$, is a rotation by angle $\Delta\theta_{k,t}$. This corresponds exactly to equation 2.9. Let us write $u_{2k,t}$ the normalized filter response (for unit vector). Now, using an elementary trigonometric identity, $\cos(a-b) = \cos(a)\cos(b) + \sin(a)\sin(b)$, the elements of this prediction matrix can be expressed as an explicit quadratic function of the normalized response:

$$\cos \Delta\theta_{k,t} = u_{2k,t}u_{2k,t-1} + u_{2k+1,t}u_{2k+1,t-1}, \text{ where } u_{2k,t} = \frac{y_{2k,t}}{(y_{2k,t}^2 + y_{2k+1,t}^2)^{1/2}}. \tag{4.2}$$

Finally, this quadratic function can be expressed in terms of squared quantities using the polarization identity: $\alpha\beta = ((\alpha+\beta)^2 - (\alpha-\beta)^2)/4$. This trick allows computing a multiplication with two squaring operations (*i.e.*, without cross terms). Specifically:

$$\begin{aligned}\cos \Delta\theta_{k,t} = \frac{1}{4}\Big(&(u_{2k,t} + u_{2k,t-1})^2 - (u_{2k,t} - u_{2k,t-1})^2 + \\ &(u_{2k+1,t} + u_{2k+1,t-1})^2 - (u_{2k+1,t} - u_{2k+1,t-1})^2\Big).\end{aligned} \tag{4.3}$$

An analogous expression can be derived for $\sin \Delta\theta_{k,t}$.

The quadratic predictor can be generalized to groups of filters of size g and expressed compactly using vector notation. The optimization objective, which can be though of as an approximate block-diagonalization, is:

$$\begin{aligned}&\min_{\mathbf{V},\mathbf{L}_1,\mathbf{L}_2} \sum_{t=1}^{T} ||\mathbf{x}_{t+1} - \mathbf{V}\mathbf{\Lambda}_t\mathbf{V}^\top\mathbf{x}_t||^2; \quad \text{where } \mathbf{\Lambda}_t = \text{blockdiag}(\mathbf{M}_{1,t}, \dots, \mathbf{M}_{K,t}), \\ &\mathbf{M}_{k,t} = \mathbf{L}_2(\mathbf{L}_1^\top\mathbf{u}_{k,t})^{\odot 2}, \quad \mathbf{u}_{k,t} = \frac{\mathbf{y}_{k,t}}{(\mathbf{1}_g^\top\mathbf{y}_{k,t}^2)^{1/2}}, \quad \text{and} \quad \mathbf{y}_t = \mathbf{V}^\top\mathbf{x}_t.\end{aligned} \tag{4.4}$$

The columns of the convolutional matrix $\mathbf{V} \in \mathbb{R}^{N\times gNK}$ contain K groups of g filters (repeated at N locations). In the quadratic predictor, the number of filters per group is not limited to pairs (this is unlike the polar prediction mechanism from equation 2.9). The prediction matrices, $\mathbf{M}_{k,t} \in \mathbb{R}^{g\times g}$, are computed adaptively as a Linear-Square-Linear cascade applied to normalized filter responses (this is a generalization of the computation in Equation 4.3). The squaring is computed pointwise and the normalized filter responses include the current and the previous time points: $\mathbf{u}_{k,t} = [u_{k,t}, u_{k+1,t}, \dots, u_{k+g-1,t}]^\top \in \mathbb{R}^g$ and $\mathbf{u}_{k,t-1}$. The linear combinations are learnable matrices $\mathbf{L}_1 \in \mathbb{R}^{2g \times d}$ and $\mathbf{L}_2 \in \mathbb{R}^{g^2 \times d}$, where d is the number of quadratic units. Note that in the case of pairs ($g = 2$), six quadratic units suffice ($d = 6$)[1].

Divisive normalization. The normalized responses of the quadratic prediction model resemble normalized simple cells (see Figure 4.2). Specifically the unit vector of responses, $u_{2k}(t)$ in equation 4.2, are linear projections of the visual input divided by the energy of related cells. This is reminiscent of the normalization behavior observed in simple cells [89, 37]. It is desirable for a prediction system to be homogeneous, that is to say the intensity of its output should scale with that of its inputs. The normalized response described here are used as multiplicative gates on principal cells, making the system altogether homogeneous. Previous models of divisive normalization put emphasis on redundancy reduction (see Figure 1.5), here we focus on its role in computing temporal predictions.

Direction selectivity. The quadratic units of the quadratic prediction model resemble the direction selective complex cells of the energy model (see Figure 4.2). Specifically the quadratic responses, the elements of the matrix $\boldsymbol{M}$ that are detailed in equation 4.3, are sensitive to temporal change in spatial phase. This selectivity for spatial displacement in a given orientation and spatial frequency is reminiscent of direction-selective complex cells which are thought to constitute the first stage of motion estimation [3, 195]. The functional mechanism for computing direction selectivity has been established in retinal circuits [15, 29] but remains unknown for primate cortex. The quadratic prediction model constitutes a hypothesis for how direction selectivity can be constructed from afferent simple cells with anti-phase spatial selectivity. The idea is that in a cortical column, nearby simple cells with similar tuning get pooled to form a direction selective complex cell. The detailed circuitry

[1] Using "Gauss's trick", each complex multiplication can be computed with only three real multiplications:

$$(a + ib)(c + id) = ac - bd + i((a + b)(c + d) - ac - bd).$$

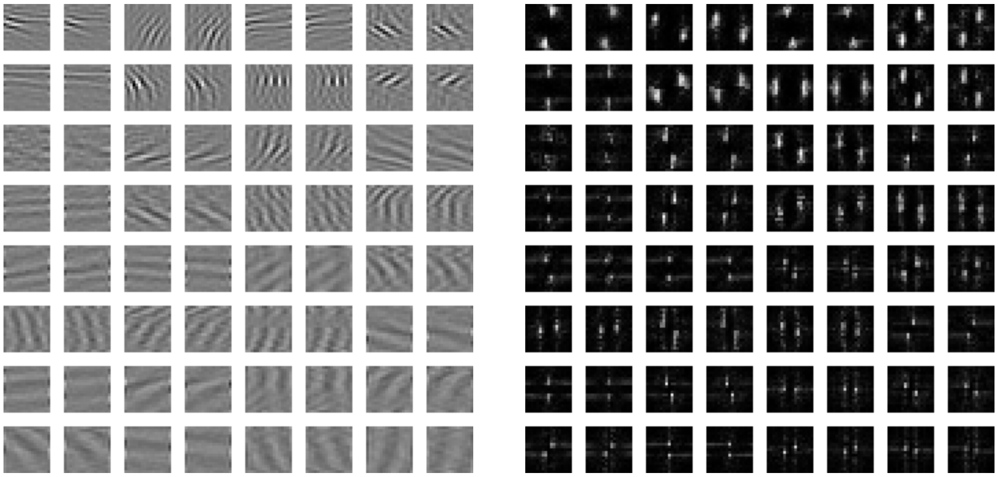


Figure 4.3: Learned quadratic predictor filters. Example filters from a Quadratic Predictor model composed of 64 pairs of filters and six quadratic units. **Left.** Spatial domain filters of size 17 × 17 pixels. The first 32 pairs are displayed side by side and sorted by frequency. Observe that the quadratic prediction mechanism for pairs also learns filters in quadrature. **Right.** Corresponding Fourier amplitude spectra displayed similarly. Observe that the filters are selective for orientation and spatial frequency.

proposed in Equation 4.3 is testable with local microcircuit anatomical measurements. Previous models of complex cells put emphasis on invariance or slowness but did not account for direction selectivity [103, 22].

4.1.2 Filter selectivity

Filters of a trained quadratic prediction model recover simple cell like selectivity with fewer constraints than the polar prediction model. A model composed of 64 pairs of channels with 6 quadratic units (*i.e.*, $K = 64$, $g = 2$ and $d = 6$) was trained for next-frame prediction on the DAVIS dataset. This model recovers the polar non-linearity and pairs of learned filters are related by a shift in spatial phase. The filters are selective of orientation and spatial frequency, as seen in Figure 4.3. The learned filters tile the frequency domain and make efficient use of limited resources. The frequency tiling is displayed with Gabor fits for individual filters in Figure 4.4. There is a greater density of filters covering low frequencies,

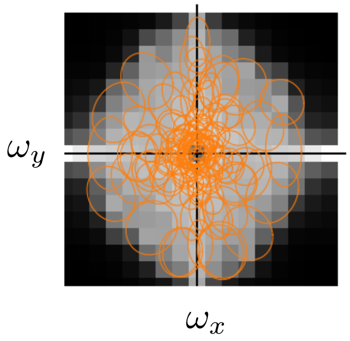


Figure 4.4: Frequency tiling. Spatial frequency selectivity of the 64 pairs of quadratic predictor filters from Figure 4.3. The two dimensional frequency domain is shown with origin at the center of horizontal and vertical frequency axes. Intensity corresponds to the superimposed power spectra of all the filters. All frequencies are equally represented, except for the highest frequencies which are ignored. For each filter, the Gaussian envelope of a Gabor fit is displayed as an orange ellipse. The population evenly tiles the frequency space and the density of filters is higher near the origin, where filters are more narrowly tuned. This corresponds to the low frequencies where visual signals have most of their energy.

which corresponds to the power law distribution of image spectra.

Beyond quadrature pairs. The quadratic prediction model generalizes polar prediction by relaxing the assumption of phase advance. Instead of building-in a phase extrapolation mechanism, prediction is calculated by a learned Linear-Squaring-Linear (or LSL) cascade. This prediction mechanism is potentially more expressive than polar prediction. A quadratic prediction model that bundles channels by groups of four was trained for next-frame prediction on the DAVIS dataset. The model achieved slightly more accurate predictions than a polar predictor or of a quadratic predictor composed of pairs of filters. The learned filters have selectivities that resemble those of a polar predictor and are displayed in Figure 4.5. The number of quadratic units in the QP mexhanism controls its expressively by varying the width of the LSL cascade.

Color. Cascading a temporal prediction module requires processing multiple channels in parallel, but we have so far only considered grayscale image sequences. As a stepping stone

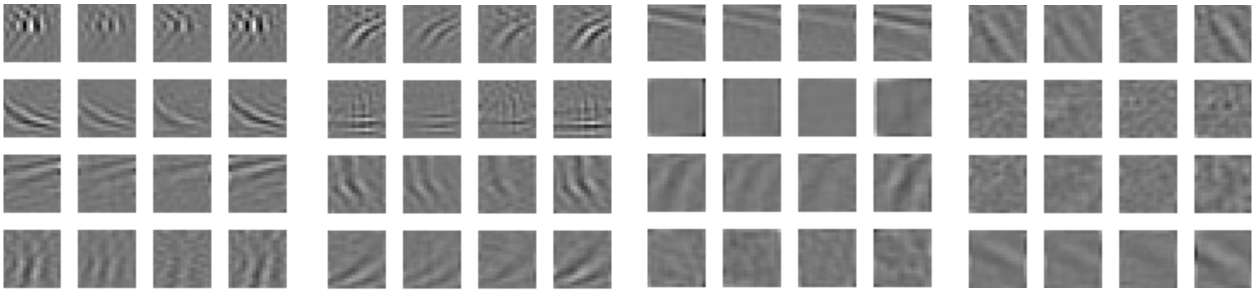


Figure 4.5: Learned groups of filters. Examples groups of filters from a quadratic predictor trained for next-frame prediction on the DAVIS video dataset. The filters form groups of 4, and the 16 groups with largest average norm out of 64 groups are shown. Notice that each groups of 4 filters are selective for orientation and spatial scale at various phases—reminiscent of polar prediction filters.

to developing a multi-stage architecture (see Section 4.2), we start by considering colored image sequences, with three channels corresponding to the RGB color model. The polar prediction model readily extends to multichannel inputs, and a model composed of 64 filters that process the three input channels in parallel was trained for next-frame prediction on the colored DAVIS dataset. The learned filters are displayed in Figure 4.6. Many of the channels resemble results for the grayscale experiment and are not color selective (luminance filters). Some channels are color selective and display red-green and yellow-blue opponency (chromaticity filters). These filters tend to have lower spatial frequency selectivity and have a matching luminance selectivity. These two categories are reminiscent of cells described in V1 [107]. The rest of the channels have more complex selectivity with a mismatch between their luminance and chromaticity selectivities (*e.g.*, first row second, forth and sixth columns).

4.1.3 Discussion

In this section, we derived the quadratic predictor, a circuit algorithm that computes polar prediction using canonical neural elements. Expressing phase extrapolation with real valued arithmetic only, the computation of temporal predictions reduces to a ratio of a cubic over a

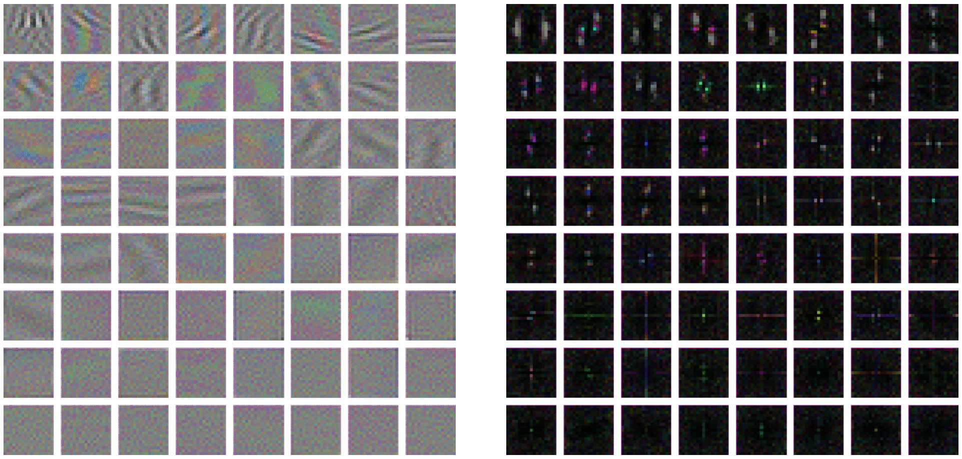


Figure 4.6: Learned color filters A polar predictor with 64 pairs of filters was trained on color videos from the DAVIS dataset. The learned color filters are shown in the spatial domain on the left and their amplitude spectrum is at corresponding localtion on the right. Filters are ordered by their euclidian norm. Only a single phase per filter is shown (this is unlike in Figure 4.1 where both filters in each pair are displayed).

quadratic that can be parameterized and learned jointly and simultaneously with the filters. This parameterized prediction mechanism generalizes the polar prediction, which is restricted to pairs of filters.

We established connections between elements of the polar prediction framework and elements of the early visual processing in primates. These connections suggest how the visual system may represents sensory inputs in a form that simplifies temporal prediction. By unifying divisive normalization and energy computation in a normative framework, the quadratic temporal prediction mechanism bridges well established aspects of visual neuroscience with the theory of temporal prediction. The polar prediction model also offers a functional explanation for perceptual straightening.

Relationship to predictive coding. Many researchers have proposed candidate algorithms for signaling predictions and prediction errors across the visual hierarchy ("predictive coding") [149, 168, 73]. This work has typically relied on Gaussian generative models and focused on the possible implementation of corresponding algorithms by neural mechanisms [105, 106]. But it is difficult to reconcile this approach with notion of selectivity, neurons responding vigorously to code for preferred feature and their selectivity increasing with learning, leading to more selective and vigorous responses, not less.

Comparatively less attention has been given to the problem of computing temporal predictions for highly non-Gaussian natural signals ("predictive processing") [90]. Our proposed polar prediction model provides a systems level hypothesis for how cortical circuits in the primate visual system might compute predictions of their inputs. Our framework is agnostic to how a circuit would use such predictions to adjust connection strengths and learn the filters. Specifically, we have relied on a standard backpropagation algorithm, only analyzing the learned solution and not the learning dynamics themselves. Local circuit mechanisms adjust synaptic weights when conflicts occur between predictions and reality. Developing a local and biologically plausible learning rule based on violation of expectation is of wide interest [69, 1, 28, 78].

Prospect for experiments. New experiments and analysis are needed to understand how perceptual straightening emerges from neural computation. The polar prediction model suggests that prediction takes place in the polar coordinates defined by simple and complex cells. This polar straightening hypothesis can be quantitatively tested through statistical modeling of neural data.

The idea of straightening neural trajectories in V1 state space has received some physiological support [93]. But there are important limitations to straightening in the standard rectangular coordinates. First, neural trajectories are constrained to evolve in a bounding box and can not stay straight for long. Indeed, neural firing rates are restricted to the positive orthant, and neurons saturate. This bounding box constraint has been shown to play an important role in forcing neural trajectories to curve [152].

A second more fundamental limitation comes from the need for trigonometric calculations

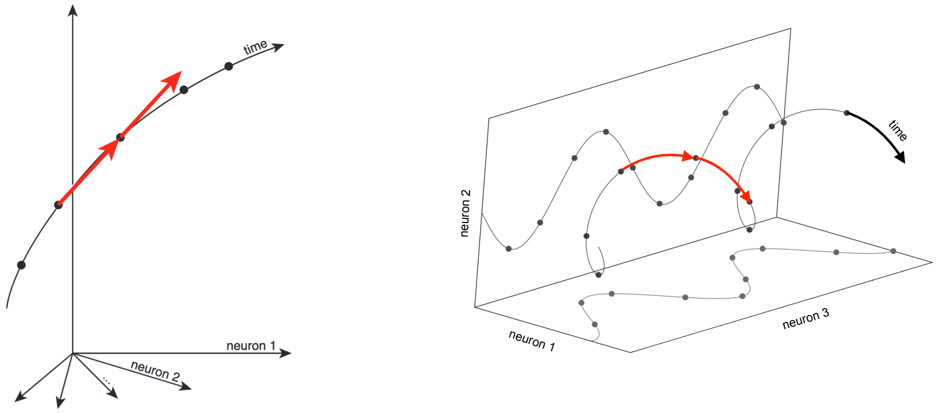


Figure 4.7: Temporal straightening. Reformulating the straightening hypothesis to compare the polar prediction model with the physiology of predictive processing. See text for discussion. **Right.** Straightening in rectangular coordinates. **Left.** Straightening in polar coordinates.

to straighten visual transformations. We showed in Section 2.2.1 that explicit straightening of the simplest transformation—translation—requires phase unwrapping and suffers from the branch cut problem (see Figure 2.6). Correspondingly it was observed that complex cells are responsible for most of rectangular straightening and that simple cells do not contribute to the reduction in curvature [93].

Polar straightening on the other hand retains the idea of linearizing trajectories but leaves it implicit along polar coordinates as illustrated by Figure 4.7. In this illustration, neuron 1 and 2 are meant to represent simple cells (each a pair of opponent cells) whose activity level oscillates with time, while neuron 3 represents a complex cell whose activity level slowly rises with time. Interestingly, rotational neural dynamics are well documented and many population level analyses of physiological recordings reveal circular neural trajectories [42, 120, 236, 75, 8].

Reanalysing this data by first characterizing units and treating simple cells and complex cells separately would allow testing the polar prediction hypothesis. Although finding simple cells and complex cells with similar selectivity and overlapping receptive fields might be challenging. Moreover, this data was collected using a controlled experimental paradigm that randomizes the order of stimulus presentation: images were flashed in random order rather than shown as continuous movies. A more natural paradigm, with image sequences shown as movies, would facilitate testing the polar prediction hypothesis.

Missing. Some aspects of the filters learned by the quadratic prediction mechanism diverge from typical V1 simple cells. In particular, the spatial extent of the filters is not well controlled in the simulations. Many filters tend to occupy the entire spatial extent of the learnable convolutional weights. Indeed these filters represent low spatial frequencies and capture the transformation of large modes of the image data. This can be alleviated by using a multiscale representation as described in section 2.3. The learned filters also contain more oscillations than typical V1 receptive fields as quantified with a Gabor fit [174].

The separation of spatial and temporal processing in the polar prediction model is at odds with the measured spatio-temporal receptive fields of neurons in V1 [49, 182]. We explored a space-time variant of the model and observed little impact on performance. This requires decoupling the analysis and synthesis filters, letting the former be spatio-temporal, while the latter remained purely spatial (analysis and synthesis filters still come in pairs). We observed that learned spatio-temporal filters are typically separable. Further analysis is required to precisely relate these results to early visual physiology.

We focused on modeling the primary visual cortex, seemingly skipping predictive processing in the retina [202, 23, 157] and in the Lateral Geniculate Nucleus (LGN) of the thalamus [55, 46]. In fact, previous work has demonstrated that both regions can indeed be well modeled as anticipating their inputs using spatial and temporal decorrelation filters. In our case, we considered video data that had already been gain adjusted, thereby reducing the need for front-end luminance and contrast gain control, which are the primary computational functions of the early visual system [138, 139].

Our approach lacks detailed mechanistic realism. For modeling neural firing rates, responses should be positive, and it would be interesting to consider opponent pairs of rectified filters in place of each unconstrained filter. More fundamentally, modeling visual temporal prediction may require considering the spikes—indeed temporal coding could be critical to enable fast processing.

Finally, incorporating gain adaptation could enable handling multiple transformations. Since circumstances are constantly changing, representations must adapt to different kinds of transformation structure. Gain adaptation represents such a rapid and reversible mechanism and it is ubiquitous in the nervous system. Modulating the amplitude of modes in the learned basis would be a natural counterpart to phase advance. The idea is to reduce prediction error by taking a gradient step with respect to the gains at each time step. This has the potential to make the polar predictor both more expressive and more neural.

4.2 Cascaded architecture for hierarchical prediction

So far, we have been focused on very short-term predictions, in the range of tens of milliseconds. Computing a detailed prediction over this short time-scale is indeed necessary to stay in tune with the rapid dynamics of our visual world. Although the details of visual signals quickly become unpredictable, extrapolating at longer temporal horizons is also of great importance. But finding temporal associations across long timescales is a notoriously challenging task.

In this section we explore the possibility of treating polar prediction as a module that can be stacked in a cascade in order to predict gradually more abstract features that are more temporally stable. Such slower variables can be integrated further back and predicted further forward. Indeed, long term prediction promotes more abstract features because those are the ones which remain coherent for longer [225]. At very long timescales, this amounts to selecting which information will be relevant and is therefore worth remembering, and to building a cognitive representation in which to plan actions [181]. Longer term prediction is also interesting because it might help understand the mystery of perceptual stability despite frequent eye movements [229]. We will describe a cascaded architecture that aims to account for the computation of stable semantic features along the visual hierarchy.

Serial prediction. Temporal prediction lends itself to subdivision. It can be broken down into a succession of largely autonomous stages that function without top-down supervision. Our hypothesis is that cascaded polar prediction modules can gradually extract slower and more abstract information. Each stage of such a cascaded model can predict at different abstraction levels and timescales by gradually factorizing its input into a part that is temporally stable and a part that evolves steadily. The idea is that each stage can be learned independently via local temporal prediction and that a slower summary is extracted and passed along to the next stage where a similar prediction architecture is applied. The extraction of slow timescales at lower levels enables learning features that evolve steadily in the next representation space—those are more abstract features. Prediction of such an internal signal is a local computation potentially amenable to a local learning rule. But visualizing and interpreting such internal signals is as not intuitive as for visual signals in the pixel domain. Conceptually we place ourselves in the position of the cortical areas downstream of V1 (*i.e.*, which only have access to neural activity in area V1).

The visual hierarchy. The anatomy of primate ventral stream is organized as a cascade of stages, each stage or visual cortical area being identified by its retinotopic map [64]. The functional properties of visual neurons become progressively more complex along this hierarchy. In the ventral stream, neural responses become gradually more selective for particular image features and more tolerant of image transformations that preserve those features [180, 52]. The receptive fields of neurons grow in spatial size with each stage of the hierarchy (roughly doubling in size from V1 to V2, and again from V2 to V4 [77]). The receptive fields are smaller and denser near the center of gaze and grow with eccentricity—each cortical retinotopic map being foveated. There is a corresponding nested hierarchy of neural timescales in the visual system [87].

Related work. Hierarchical models of visual processing have a long history and are typically composed of cascaded convolutional filtering applied to energy like signals [83, 74]. The role of depth in such cascades has been linked to efficiently representing visual signals with compositional structure [6, 7]. Cascaded statistical models of images patches, which are motivated by efficient coding, have been compared to the selectivity of neurons in area V2 [99]. Hierarchical temporal predictive coding models have also been developed [105, 106], and hierarchical temporal prediction was used as a candidate model for the dorsal visual stream [198]. Note however that in those studies, the convolution-rectification architectures used are not well suited to capture the multiplicative computations required for temporal prediction (see Section 2.1.2).

4.2.1 A two-stage model

The polar prediction model described in Chapter 2 and the quadratic prediction model of Section 4.1 both assume that amplitudes are sufficiently smooth to be left constant by the prediction mechanism. Indeed, learned pairs of filters have linearly advancing phases and slow amplitudes. But these slow amplitudes still contain abundant structured change which is partially predictable. This suggests cascading the prediction module onto the amplitudes as depicted in Figure 4.8. For simplicity, we will fix the first stage, W_1, to a steerable pyramid and only learn the second stage, W_2.

Setup. A second stage polar predictor model was trained to predict next frame steerable pyramid amplitudes on the DAVIS dataset. The pyramid is composed of four orientations

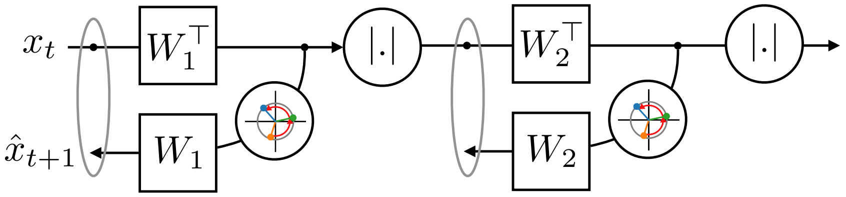


Figure 4.8: Two-stage architecture. A multi-stage model aiming to gradually factorize stability and change. Image sequences are spatially filtered by a first stage, $W_1^\top$, and future coefficients are predicted via phase advance and projected back to the input space, W_1. This first stage is trained by minimizing prediction mean squared error in the input space. Amplitudes from the first stage coefficients are more stable than the input image sequence but still contain predictable structure. The second stage processes these amplitudes with the same polar prediction architecture. The prediction objective measured layerwise, phase stays local within each stage and only amplitudes are propagated forward. Note that second layer spatial filters, $W_2^\top$, are applied to a multichannel input stream.

at three scales, with even and odd symmetric filters as shown in Figure 1.4—the high and low pass residuals are ignored. The amplitudes are computed by combining even and odd responses and then spatially downsampled by a factor two. The second layer convolutional filters are of size 11 × 11 pixels and span all 12 input channels. This second stage contains 64 pairs of filters. There is an intermediate normalization layer between steerable pyramid amplitudes and the second layer filters. This batch normalization layer enforces that the input to the second layer has mean zero and variance one in each of the 12 input channels. There is a small penalty on the euclidian norm of the weights (weight decay $\lambda = 10^{-2}$).

Preliminary Results. Three example learned filters are illustrated in Figure 4.9. Most of the filters compute motion integration across orientation and scale, two such filters are examined in more detail in Figure 4.10. This corresponds to two-stage models of visual motion representation, where velocity is extracted from linear combinations of spatio-temporal energy [195]. Some of the remaining channels (not shown) are selective for global luminance changes, for orientation changes, and for scale changes. The rest of the channels, about a fourth, are less intuitively interpretable.

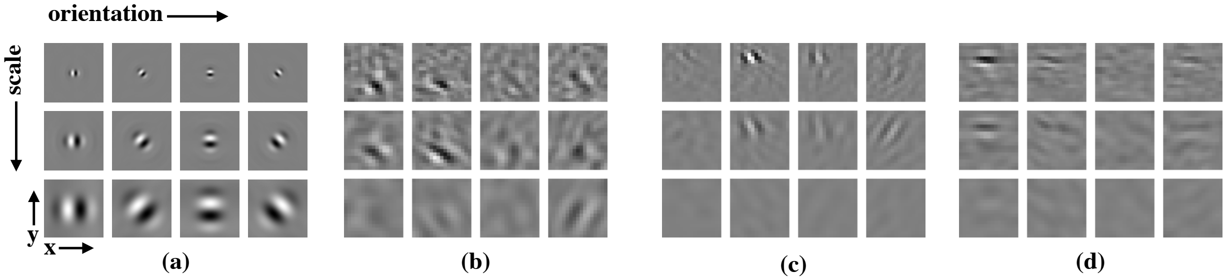


Figure 4.9: Second layer filters. Three example second layer filters from a two-stage architecture trained for next frame prediction on the DAVIS dataset. **(a)** First stage steerable pyramid filters at four orientations and three scales, only odd symmetric filters shown, displayed for reference. Pyramid filters define the input space on which the second layer operates (specifically the amplitude coefficients) and second layer filters in the following pannels are displayed using the same layout. **(b)** Obliquely oriented filter selective for amplitude in oblique orientation at medium scale. **(c)** Vertically oriented filter selective for amplitude in oblique orientation at fine scale. **(d)** Horizontally oriented filter selective for amplitude in vertical orientation at fine scale.

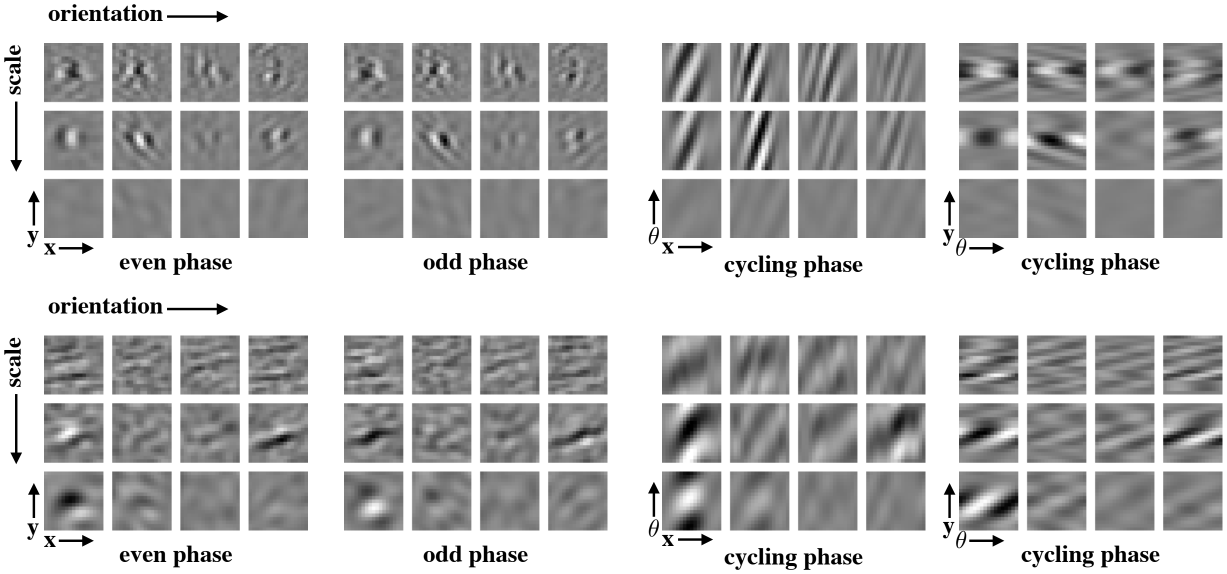


Figure 4.10: Two example filters. Second layer filters that integrate coherent spatio-temporal amplitude across orientation and scale. **Top.** Vertically oriented filter selective for amplitude in vertical orientation at fine scales. Filter pair shown in rectangular coordinates on the left. Cycling though phases produces spatio-temporal filters and vertical and horizontal slices are displayed on the right. Observe that the filter pair is selective for coherent left-right motion. **Bottom.** Horizontally oriented filter selective for amplitude in vertical orientation at coarse scale. The filter pair is selective for coherent down-up motion.

4.2.2 Discussion

In this section we constructed a hierarchical model by treating polar prediction as a module that can be cascaded. The model suggests a computational role for complex cells in each layer: to extract representations that are "stable to dynamics", by factorizing out the temporally stable information [17]. This model can serve as a guide to explain mid-level vision, especially the extraction of abstract features at slow time scales from low level measurements. We hypothesize that the transformations of the visual stream arise through an unsupervised learning process that aims at facilitating temporal prediction, one stage at a time.

Why layerwise? End-to-end back-propagation is generally considered biologically implausible. The two-stage prediction model described here relies only on local computations only, and those local computations are analogous to the lateral dynamics of cortical neurons. Each layer is learned by predicting its input, which does not require image reconstruction in the pixel domain (except for the first stage). There is no feedback between stages, which avoids the representation collapse problem common to many layerwise representation learning schemes. More importantly, the layerwise temporal prediction objective provides a learning signal that constrains all the intermediate stages of the representation. This is unlike end-to-end learning, which often constrains the overall network function while leaving intermediate stages under-determined. Previous work explored a related strategy for representation learning with layerwise self-supervised learning objectives [158]. That study was restricted to processing static images as opposed to image sequences, and it required the use of different inputs to train each layer (images of different size). In the method we presented, the temporal hierarchy can automatically match task complexity with the expressivity of each layer.

This is analogous to the extraction of gradually more abstract representations in the successive stages of a scattering transform [137, 32]. The scattering transform extracts a representation that is stable to deformations by computing a cascade of wavelet decomposition and amplitudes non-linearity, and extracting a spatially averaged amplitude per channel at each stage. Stability to deformation is similar to temporal stability when the signal dynamics are dominated by deformations. But unlike our proposed model, at each stage of the scattering transform, a wavelet decomposition is computed separately for each channel. Such a tree structure is problematic because it does not capture interactions between channels and because the number of channels grows exponentially with depth (although some pruning is possible). Our proposed model learns a scattering-like representation with two important differences: i) only amplitudes are propagated forward, but the model also uses phases, those phases are used for computing temporal prediction and stay local within each stage, and ii) each successive stage pools information across its input channels, *i.e.*, there are lateral interactions, which make the model more expressive than traditional tree-structured ones.

Missing. The qualitative interpretation of example second layer filters does not give a full understanding of their selectivity. Generating an image sequence that is maximally activating would demonstrate what aspects of the visual scene are captured by a second-layer filter. The hierarchical construction proposed in this section was motivated by the idea of extracting more abstract and stable representations. This can be tested explicitly by quantifying the temporal stability of successive stages. Correspondingly, slower modes should enable longer-term prediction. Moreover, the degree of abstraction of the learned features can be tested by quantifying how transferable the representation is for downstream tasks such as scene segmentation, heading direction prediction, or object recognition. This analysis can be repeated for models composed of one, two, or more polar prediction stages.

The two-layer temporal prediction model makes predictions for visual processing downstream of area V1. By some aspect it is similar to area MT, by others it is similar to area

V2. Comparing the model to physiological recordings in these area would help refine our hypotheses. In particular, the second layer simple-like units pool compatible orientation energy from the first layer, which is reminiscent of motion selective units in area MT [233]. Temporal prediction could help explain how such selectivity could be learned, a question that was left open by visual motion processing models of area MT [195]. The second layer complex-like units compute feature correlations across space, orientation, scale and phase, which is reminiscent of texture selective units in area V2 [239]. Moreover, the learned filters of our two layer model are compatible with the analysis of neural selectivity in area V2 [153]. These V2-like complex units compute auto- and cross-correlations across local oriented energy of a multiscale representation, which corresponds to texture statistics [162].

4.3 On what the visual system is computing

Physiological recordings afford partial access to what is represented by the visual system. But the use of these representations has to be inferred in order to understand what is being computed. This requires describing the transformation of representations from one area to the next. The theory of efficient coding and of optimal inference can help inform this process. Similarly, the theory of temporal prediction can also guide model development: asking how the brain computes the future helps constraining our hypotheses. In this section, we discuss the physiological and perceptual phenomena that our models seek to explain, as well as corresponding experiments that would help testing and refining these models.

4.3.1 Physiology: functional organization of visual cortex

The speed of sight. Processing visual information in real time is challenging, especially in rapidly changing environments. The constraints of online processing demand fast communication of neural signals over long cortical distances. Electric spikes are actively propagated along axons by voltage-gated channels that send information quickly. Yet axonal conduction delays remain significant as the speed of propagation stays under 100 meters per second [220]. Moreover, the response latency of post-synaptic neurons adds to processing delay, and neuronal integration time depends on the state of the membrane.

It is remarkable that despite these biophysical delays visual categorization can be computed very rapidly. In primates, a single volley of spikes takes about 70 ms to propagate through the visual hierarchy up to area IT and is sufficient to support rapid visual categorization [206]. The onset latency of neurons along the visual hierarchy increases regularly, adding about 10 ms per stage [151]. It is notable that area MT, which is selective for motion, responds faster than V2, which is selective for visual texture.

Moreover the spread of neural latencies within visual areas tends to be larger than the difference of latencies across areas. This may be due to lateral interactions within each area, where local recurrent circuits compute and transform representations. These local circuits are more dense, rapid and precise that cross area projections and are therefore a natural candidate for the computation of temporal prediction. These circuits also continually adapt to changes in the environment, and also adjust as they operate so as to be resilient to changes in underlying biological implementation (*e.g.*, robustness to neural damage). Our polar prediction model constitutes a hypothesis for utilizing and continually learning such fast local computation.

Functional anatomy. Although many hierarchical models of visual processing assume that only V1 complex cells are projected to V2, evidence from precise physiological experiments suggests otherwise [58]. There is partial evidence that MT receives primarily complex cell afference from V1. Overall, we need more systematic measurements of the functional properties of V1 projections into the dorsal and ventral streams.

There is ample physiological evidence for existence of feedback connections across visual cortical areas, but their functional role remains difficult to characterize. Feedback projections from V2 to V1 seem responsible for late texture modulation in the primary visual cortex [240]. Fast prefrontal recurrent processing and projection from the ventro-lateral prefrontal cortex contribute to object recognition [114, 113]. Better understanding of these processes requires developing models with feedback. In this regard, it would be interesting to study the functional contribution of higher stages to condition temporal prediction in lower stages.

The broad division of the visual system into ventral and dorsal streams is well established [208]. The dorsal pathway (V1, V3, V5) represents position, motion, and action; while the ventral pathway (V1, V2, V4, IT) represents spatial form, recognition, and leads to memory. But the interactions between these parallel streams, although well documented [123, 124, 224], remains poorly understood. Because temporal prediction requires the combination of motion and form, temporal prediction models hold the potential of providing a normative framework to study these interactions.

4.3.2 Perception: seeing what comes

Gestalt. It has long been recognized that our perceptual systems exploit cues available in the signal by grouping together elements in the visual scene that have a continuous contour or that move together. The Gestalt school of psychology described “continuity”, “common fate” and other such laws. Understanding the origin of these disparate laws and how they come together on natural visual signals is of interest. Perceptual effects such as the illusory shift in the location of a patch containing visual motion reveal the predictive nature of visual processing. This motion-induced position shift [164] reflects the visual system’s anticipation of the future position of moving image content. Deriving a synthetic visual system by training for temporal prediction of dynamic visual scenes constitutes a promising approach for learning “intuitive physics” [20].

Development. In humans, the acquisition of spatio-temporal object constancy occurs within the first four months of life—before infants can learn through motor interaction with their environment [115]. Partially occluded objects that move coherently across the occluder are perceived as unitary. It has been suggested that infants begin life with fundamental notions like coherence, unitarity and persistence. In turn these notions enable infants to temporally predict the future position of moving objects [201]. The temporal prediction networks described in this thesis provide a basis for the computational understanding of these fundamental notions, and of how they are combined with sensory information. The critical elements of the network architecture, local gain control and squaring, constitute a candidate hypothesis for the structures that may be present at birth and then refined during development.

Temporal masking. Visual perception is lossy. Many aspects of the visual scene are not perceived, and certain features tend to take priority over others. For example, fast motions are a powerful cue for objectness and parts of the scene moving together tend to be grouped together. Moreover, moving objects typically do not also morph, change size, or color at the same time as they move. Correspondingly, our visual system seems to expect that moving objects do not change form. This is revealed by a striking temporal masking illusion: objects changing in color, luminance, size, or shape appear to stop changing when they move [205]. This striking phenomenon, referred to as event silencing, exemplifies a form of perceptual interference between two features of a stimulus, with fast motion dominating other features. Temporal prediction might be able to account for such phenomena, and networks trained on next frame prediction can be use to test this hypothesis. Indeed, studying the implicit motion prior embedded in prediction networks can help explain perceptual illusions [219].

Perceptual stability. Although we actively move our eyes many times per second, our perceptual experience of the environment is stable, suggesting that our visual system integrates information across saccades [76]. One common hypothesis for the coordination of visual information over space and time is that of shifter circuits [5]. This model proposes an explicit copy-pasting mechanism that registers visual inputs onto some internal perceptual space. This idea is largely inspired from motion compensation in video processing, where it has been successfully implemented. But the biological plausibility of this method has been questioned. Indeed, handling arbitrary image shifts comes with a prohibitive wiring cost. Even more problematic is the absence of regular grids in visual cortex that could be registered with one another. Instead, topographic maps are foveated, with denser sampling of the central region of gaze and magnified receptive fields in the periphery.

Foveated visual representations combine high resolution at the center of gaze with broad contextual information in the periphery. The latter guides saccades to relevant visual locations, while the former enables analysis of fine details at those select locations. Tracking object motion stabilizes a target onto the fovea but blurs the background. Correspondingly, local and global visual motion cues are dominant in the periphery and the fovea respectively [88]. Peripheral vision is subject to crowding, with visual features summarized and represented by a texture-like code [71, 177, 238]. The gradual increase in receptive field size allows pooling information spatially and extracting stable features along the ventral hierarchy—which may help explain how sensory information is integrated into a unified and yet detailed percept. This hypothesis for the origin of perceptual stability may seem to be at odds with the fact that the resolution of our perceptual experience only matches the resolution of V1 cells, and not that of IT for example. But pooling does not have to blur features and limit perceived resolution, instead it can extract invariant information along well defined features. This is best illustrated with a familiar example: a complex cell is more invariant to position than a matched simple cell, yet it has the same spatial resolution and is just as finely tuned to frequency.

Chapter 5

Discussion

In this thesis, we explored the topic of temporal prediction and proposed that it can serve as a unifying theory for visual processing. We emphasized local neural computations, factorized representations, and probabilistic inference on dynamic visual signals.

In Chapter 2, we proposed a model for discovering and exploiting the predictable structure in image sequences. The model factorizes transformations with a simple polar mechanism. We demonstrated how this multiplicative architectural building block enables visual temporal prediction.

Future work will aim to extend the expressivity of the model by stacking polar modules in a multi-layered architecture. Such an architecture could discover groups from partial observation (*e.g.*, 2d projection of 3d transformation) and jointly represent multiple groups. Of particular interest is the study of symmetry discovery beyond commutative groups. There are also potential applications of the polar prediction model to engineering tasks such as video compression, where the development of flow-free methodologies could have important practical benefits.

In the following chapter, Chapter 3, we developed a probabilistic framework for temporal prediction that that can handle ambiguity and cope with incomplete information in high-dimensions. We optimized a model for noise resilience and used it to sample probable next frames in image sequences. This model adaptively combines past conditioning and noisy observation.

Future work will aim to elucidate how trained conditional denoisers compute predictions, and the polar prediction model could serve as a helpful analysis tool. Further work will distill the sequence-to-image denoiser architecture into an explicit representation that is both prior informed and abstract.

In the last chapter, Chapter 4, we discussed the relevance of polar prediction for biological modeling. We mapped the model to canonical sensory computations that have received ample evidence. We also outlined how to extract more abstract features by cascading the polar mechanism.

Future work will design and analyze physiological and perceptual experiments to quantitatively test the polar straightening hypothesis. The hierarchical model outlined in this chapter should be further developed to handle prediction at longer timescales and serve as a candidate model for mid-level visual areas. Finally, bringing the probabilistic framework of Chapter 3 closer to biology might offer a way to think about how sensory systems adapt to (*i.e.*, learn, store, update, and utilize) environmental probabilities.

Appendix A

Notations

Let $\mathbb{R}^N$ denote the N-dimensional Euclidean space equipped with the usual Euclidean norm $||\cdot||$ and let $\mathbb{C}^K$ denote the K-dimensional complex vector space. Let $\mathbb{C}^{N\times K}$ denote the set of $N \times K$ complex-valued matrices. Let $\mathbf{F}^*$ denote the adjoint (ie. the conjugate transpose) of $\mathbf{F}$.

Given vectors $\mathbf{u}, \mathbf{v} \in \mathbb{C}^K$, let $\mathbf{u} \odot \mathbf{v} = [u_1 v_1, \ldots, u_K v_K]^\top \in \mathbb{C}^K$ denote the elementwise (aka. Hadamard) product of $\mathbf{u}$ and $\mathbf{v}$. Let $\mathbf{u} \oslash \mathbf{v} = [u_1/v_1, \ldots, u_K/v_K]^\top \in \mathbb{C}^K$ denote the elementwise division of $\mathbf{u}$ and $\mathbf{v}$. Let $\mathrm{diag}(\mathbf{u})$ denote the $K \times K$ diagonal matrix whose $(k,k)^{th}$ entry is u_k.

Let the complex number $z \in \mathbb{C}$ be expressed in rectangular coordinates as $z = x + iy$ where $(x, y) \in \mathbb{R}^2$, or in polar coordinates as $z = ae^{i\theta}$ where $(a, \theta) \in \mathbb{R}_+ \times [-\pi, \pi]$. Its complex conjugate is $\overline{z} = x - iy = ae^{-i\theta}$. The rectangular coordinates are $u = a\cos(\theta)$ and $v = a\sin(\theta)$; and the polar coordinates are $a = |z| = \sqrt{x^2 + y^2}$, and $\theta = \angle z = \mathrm{atan2}(y, x)$. We overload these notations to denote element-wise operations on a vector $\mathbf{z} \in \mathbb{C}^K$: $\overline{\mathbf{z}} = [\overline{z_1}, \ldots, \overline{z_K}]^\top \in \mathbb{C}^K$ and $|\mathbf{z}| = [|z_1|, \ldots, |z_K|]^\top \in \mathbb{R}_+^K$.

Appendix B

Representing visual transformations

Description of architectures

Motion Compensation. We compare our method to the traditional motion-compensated coding approach that forms the core of inter-picture coding in well established compression standards such as MPEG. Block matching is an essential component of these standards, allowing the compression of video content by up to three orders of magnitude with moderate loss of information. For each block in a frame, typical coders search for the most similar spatially displaced block in the previous frame (typically measured with MSE), and communicate the displacement coordinates to allow prediction of frame content by translating blocks of the (already transmitted) previous frame. We implemented a “diamond search” algorithm [237] operating on blocks of 8 × 8 pixels, with a maximal search distance of 8 pixels which balances accuracy of motion estimates and speed of estimation (the search step is computationally intensive). We use the estimated displacements to perform causal motion compensation (**cMC**), using displacement vectors estimated from the previous two observed frames ($\mathbf{x}_{t-1}$ and $\mathbf{x}_t$) to predict the *next* frame ($\mathbf{x}_{t+1}$) rather than the current one (as in MPEG).

Complex Steerable Pyramid. We consider a fixed multiscale oriented representation of image content: a steerable pyramid [194, 162] covering 16 orientations and 5 scales on the DAVIS dataset (resp. 16 orientations and 4 scales on VanHateren dataset). This choice of number of orientations and number of scales maximizes prediction performance on the corresponding datasets.

Polar Predictor. We use 32 pairs of convolutional channels with filters of size 17 × 17 pixels, without biases (no additive constants) and no padding (ie. "valid" boundary condition). For the multiscale version (mPP), we use 16 pairs of convolutional channels with filters of size 11 × 11 pixels, without biases (no additive constants). The representation is computed inside a fixed Laplacian pyramid [34]. We used 4 scales for the DAVIS dataset (and respectively 3 scales for the VANH dataset). Within this multiscale representation, the learned filters are applied with zero padding (ie. "same" boundary condition).

Vanilla CNN. Finally, we implemented a more direct convolutional neural network predictor (**CNN**), that maps two successive observed frames to an estimate of the next frame [143]. For this, we used a CNN composed of 20 stages, each consisting of 64 channels, and computed with 3 × 3 filters without additive constants, followed by half-wave rectification (ie. ReLU). To facilitate learning, a skip connection copies the current frame $\mathbf{x}_t$ and the network only outputs residuals that get added to the current frame in order to predict the next frame: $\hat{\mathbf{x}}_{t+1} = \mathbf{x}_t + f_w([\mathbf{x}_t, \mathbf{x}_{t-1}])$. This model jointly transforms and processes pairs of frames to generate predictions, while both polar predictor (PP) and quadratic predictor (QP) separate spatial processing and temporal extrapolation.

U-net. This multi-resolution architecture is commonly used for image-to-image tasks [176]. It has an analysis-synthesis structure with downsampling, upsampling, and skip connections between levels at the same resolution. We used 5 stages, each containing a computational block that consists of two convolutions, batch norm and rectification. The convolutions have filters of size 3 × 3, comprise 64 channels, and no additive bias. The end-to-end network comprises 20 non-linear stages with ReLU activations.

Description of datasets and optimization

DAVIS. To train, test and compare these models, we use the DAVIS dataset [161], which was originally designed as a benchmark for video object segmentation. Image sequences in this dataset contain diverse motion of scenes and objects (eg., with fixed or moving camera, and objects moving at different speeds and directions), which make next frame prediction challenging. Each clip is sampled at 25 frames per second, and is approximately 3 seconds long. The set is subdivided into 60 training videos (4741 frames) and 30 test videos (2591 frames). We pre-processed the data, converting all frames to monochrome luminance values, and scaling their range to the interval $[-1, 1]$. Frames are cropped to a 256×256 central region, where most of the motion tends to occur, and then spatially down-sampled to 128×128 pixels.

VanHateren. We also consider a smaller video dataset consisting in footage of animals in the wild [211] which contains a variety of motions (animals in the scene, camera motion, etc.) and occlusions. The missing band at the top the frame is cropped, reducing the image size from 128×128 pixels to 112×128 pixels. The dataset is standardized to zero mean and unit variance, it is split into snippets of 11 frames, 292 snippets are used for training and 33 for testing. There is no spatial downsampling or whitening.

Boundary handling. The computation of this prediction error is restricted to the center of the image because moving content that enters from outside the video frame is inherently unpredictable. Specifically, we trim a 17-pixel strip from each side, yielding frames of size

94×94 pixels for the DAVIS dataset and 78×94 for the VanHateren dataset. Convolutions are computed with zero padding so that the outputs have the same size as inputs.

Training procedure. We assume the temporal evolution of natural signals to be sufficiently and appropriately diverse for training, and do not apply any additional data augmentation procedures. We train on brief temporal segments containing 11 frames (which allows for prediction of 9 frames), and process these in batches of size 4. We train each model for 200 epochs on DAVIS using the Adam optimizer [117] with default parameters and a learning rate of $3 \cdot 10^{-4}$. The learning rate is automatically halved when the test loss plateaus. In the CNN, we use batch normalization before every half-wave rectification, rescaling by the standard deviation of channel coefficients (but with no additive bias terms).

Appendix C

Estimating optimal predictors

Description of architectures

The conditional denoisers considered here are all three scales U-nets [176]. The network takes as input the past conditioning frames concatenated with the noisy observation. The number of past conditioning frame is controlled by varying the number of input channels. The filters are of size 3×3, the first stage contains 64 filters. Each successive stage of the encoder downsamples the image by a factor 2 in each spatial dimension and doubles the number of channels. Each stage of the decoder reverses this, upsampling spatially, and halving the number of channels–while combining information across stages in a coarse-to-fine manner. The network outputs an estimated next frame, there is no input-output skip connection, this is unlike the one used in Appendix 5.

Description of datasets and optimization

Moving Leaves. Each image sequence contains two disks on a blank background that move and occlude each other. In each sequence, both disks have the same physical size but their distance to the imaging plane is randomized. As a result, each disk's projected 2d size is a reliable, although indirect, cue to its distance in the scene. The disk with larger projected 2d size always occlude the smaller one. The trajectories of each disk are sampled from Gaussian processes, and their average speed of motion scales inversely with distance. This motion parallax is an additional, albeit weaker cue to a disk's distance.

This synthetic dataset contains 10^5 image sequences, and is split into train and test set with at a 9 to 1 ratio. Each sequence is composed of 11 frames and is of size 32×32 pixels. For each image sequence, luminance and depth are sampled randomly and uniformly. The spatial positions (x and y) are sampled from a Gaussian process, and the bandwidth of the GP scales with depth (smoother slower trajectories for objects further away). In each image sequence, there is at least one frame where half of smaller disk is occluded by the larger disk.

DAVIS32. Small image sequences were cropped out of the DAVIS dataset, this procedure is different from the one described in Appendix 5. This dataset contains about $5 \cdot 10^4$ small image sequences, divided into about 36 thousands training sequences (or about 400 thousand frames), and about 5 thousand test sequences (or about 60 thousand frames). Each image sequence is composed of 11 frames of size 32×32 pixels. The spatial crops of the large DAVIS images are taken at 21 spatial locations and 3 different scales. The data is rescaled so that grayscale intensity lies in $[0, 1]$.

Training procedure. The networks are trained to minimise mean squared denoising error, and the procedure is almost identical to that used in Appendix 5. The only differences is that the networks are trained for 1000 epochs. Moreover, the square root of the noise level is sampled uniformly between 0 and 1.

Bibliography

[1] Laurence F Abbott and Kenneth I Blum. Functional significance of long-term potentiation for sequence learning and prediction. *Cerebral cortex*, 6(3):406–416, 1996. 89

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 4, 41

[3] Edward H Adelson and James R Bergen. Spatiotemporal energy models for the perception of motion. *Josa a*, 2(2):284–299, 1985. 13, 82, 84

[4] Shun-ichi Amari. Invariant structures of signal and feature spaces in pattern recognition problems. *RAAG Memoirs*, 4(1-2):553–566, 1968. 79

[5] Charles H Anderson and David C Van Essen. Shifter circuits: a computational strategy for dynamic aspects of visual processing. *Proceedings of the National Academy of Sciences*, 84(17):6297–6301, 1987. 103

[6] Fabio Anselmi, Joel Z Leibo, Lorenzo Rosasco, Jim Mutch, Andrea Tacchetti, and Tomaso Poggio. Unsupervised learning of invariant representations in hierarchical architectures. *arXiv preprint arXiv:1311.4158*, 2013. 94

[7] Fabio Anselmi, Lorenzo Rosasco, and Tomaso Poggio. On invariance and selectivity

in representation learning. *Information and Inference: A Journal of the IMA*, 5(2): 134–158, 2016. 94

[8] Mikio C Aoi, Valerio Mante, and Jonathan W Pillow. Prefrontal cortex exhibits multidimensional dynamic encoding during decision-making. *Nature neuroscience*, 23(11): 1410–1420, 2020. 90

[9] Hasan F Ates and Michael T Orchard. Spherical coding algorithm for wavelet image compression. *IEEE Transactions on Image Processing*, 18(5):1015–1024, 2009. 24

[10] Fred Attneave. Some informational aspects of visual perception. *Psychological review*, 61(3):183, 1954. 5

[11] Omri Azencot, N Benjamin Erichson, Vanessa Lin, and Michael Mahoney. Forecasting sequential data using consistent koopman autoencoders. In *International Conference on Machine Learning*, pages 475–485. PMLR, 2020. 21

[12] Mohammad Babaeizadeh, Chelsea Finn, Dumitru Erhan, Roy H Campbell, and Sergey Levine. Stochastic variational video prediction. In *International conference on machine learning*. PMLR, 2018. 51

[13] Louis Bachelier. Théorie de la spéculation. In *Annales scientifiques de l'École normale supérieure*, volume 17, pages 21–86, 1900. 4

[14] Manel Baradad Jurjo, Jonas Wulff, Tongzhou Wang, Phillip Isola, and Antonio Torralba. Learning to see by looking at noise. *Advances in Neural Information Processing Systems*, 34:2556–2569, 2021. 63

[15] HB Barlow and William R Levick. The mechanism of directionally selective units in rabbit's retina. *The Journal of physiology*, 178(3):477, 1965. 84

[16] Horace Barlow. Redundancy reduction revisited. *Network: computation in neural systems*, 12(3):241, 2001. 13

[17] Horace Barlow. Linking minds and brains. *Visual Neuroscience*, 30(5-6):207–217, 2013. 97

[18] Horace B Barlow. Unsupervised learning. *Neural computation*, 1(3):295–311, 1989. 6

[19] Horace B Barlow et al. Possible principles underlying the transformation of sensory messages. *Sensory communication*, 1(01):217–233, 1961. 5

[20] Peter W Battaglia, Jessica B Hamrick, and Joshua B Tenenbaum. Simulation as an engine of physical scene understanding. *Proceedings of the National Academy of Sciences*, 110(45):18327–18332, 2013. 101

[21] Yoshua Bengio, Réjean Ducharme, and Pascal Vincent. A neural probabilistic language model. *Advances in Neural Information Processing Systems*, 13, 2000. 52

[22] Pietro Berkes and Laurenz Wiskott. Slow feature analysis yields a rich repertoire of complex cell properties. *Journal of vision*, 5(6):9–9, 2005. 85

[23] Michael J Berry II, Iman H Brivanlou, Thomas A Jordan, and Markus Meister. Anticipation of moving stimuli by the retina. *Nature*, 398(6725):334, 1999. 91

[24] Matthias Bethge, Sebastian Gerwinn, and Jakob H Macke. Unsupervised learning of a steerable basis for invariant image representations. In *Human Vision and Electronic Imaging XII*, volume 6492, pages 132–143. SPIE, 2007. 44

[25] William Bialek, Ilya Nemenman, and Naftali Tishby. Predictability, complexity, and learning. *Neural computation*, 13(11):2409–2463, 2001. 13

[26] Fischer Black and Myron Scholes. The pricing of options and corporate liabilities. *Journal of political economy*, 81(3):637–654, 1973. 4

[27] David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017. 51

[28] Wieland Brendel, Ralph Bourdoukan, Pietro Vertechi, Christian K Machens, and Sophie Denéve. Learning to represent signals spike by spike. *PLoS computational biology*, 16(3):e1007692, 2020. 89

[29] Kevin L Briggman, Moritz Helmstaedter, and Winfried Denk. Wiring specificity in the direction-selectivity circuit of the retina. *Nature*, 471(7337):183–188, 2011. 84

[30] Michael M Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*, 2021. 44

[31] T Brooks, B Peebles, C Homes, W DePue, Y Guo, L Jing, D Schnurr, J Taylor, T Luhman, E Luhman, et al. Video generation models as world simulators, 2024. 56

[32] Joan Bruna and Stéphane Mallat. Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1872–1886, 2013. 97

[33] Arthur Earl Bryson and Yu-Chi Ho. *Applied Optimal Control: Optimization, Estimation, and Control*. Blaisdell, Waltham, MA, 1969. 4

[34] Peter Burt and Edward Adelson. The laplacian pyramid as a compact image code. *IEEE Transactions on communications*, 31(4):532–540, 1983. 10, 37, 108

[35] Charles F Cadieu and Bruno A Olshausen. Learning intermediate-level representations of form and motion from natural movies. *Neural computation*, 24(4):827–866, 2012. 18, 45, 80

[36] Matteo Carandini and David J Heeger. Normalization as a canonical neural computation. *Nature reviews neuroscience*, 13(1):51–62, 2012. 77

[37] Matteo Carandini, David J Heeger, and J Anthony Movshon. Linearity and normalization in simple cells of the macaque primary visual cortex. *Journal of Neuroscience*, 17(21):8621–8644, 1997. 84

[38] Ho Yin Chau, Frank Qiu, Yubei Chen, and Bruno Olshausen. Disentangling images with lie group transformations and sparse coding. *arXiv preprint arXiv:2012.12071*, 2020. 45

[39] Gal Chechik, Amir Globerson, Naftali Tishby, and Yair Weiss. Information bottleneck for gaussian variables. *Journal of Machine Learning Research*, 6(Jan):165–188, 2005. 14

[40] Yubei Chen, Dylan Paiton, and Bruno Olshausen. The sparse manifold transform. *Advances in Neural Information Processing Systems*, 31, 2018. 45

[41] Bilal Chughtai, Lawrence Chan, and Neel Nanda. A toy model of universality: Reverse engineering how networks learn group operations. In *International Conference on Machine Learning*, pages 6243–6267. PMLR, 2023. 44

[42] Mark M Churchland, John P Cunningham, Matthew T Kaufman, Justin D Foster, Paul Nuyujukian, Stephen I Ryu, and Krishna V Shenoy. Neural population dynamics during reaching. *Nature*, 487(7405):51–56, 2012. 90

[43] Taco Cohen and Max Welling. Learning the irreducible representations of commutative lie groups. In *International Conference on Machine Learning*, pages 1755–1763. PMLR, 2014. 45

[44] Taco Cohen and Max Welling. Group equivariant convolutional networks. In *International conference on machine learning*, pages 2990–2999. PMLR, 2016. 44

[45] Kenneth James Williams Craik. *The nature of explanation*, volume 445. Oxford, England: University Press, Macmillan, 1943. 4

[46] Yang Dan, Joseph J Atick, and R Clay Reid. Efficient coding of natural scenes in the lateral geniculate nucleus: experimental test of a computational theory. *Journal of neuroscience*, 16(10):3351–3362, 1996. 91

[47] John Daugman. Statistical richness of visual phase information: update on recognizing persons by iris patterns. *International Journal of computer vision*, 45:25–38, 2001. 24

[48] Peter Dayan. Improving generalization for temporal difference learning: The successor representation. *Neural computation*, 5(4):613–624, 1993. 4

[49] Gregory C DeAngelis, Izumi Ohzawa, and Ralph D Freeman. Receptive-field dynamics in the central visual pathways. *Trends in neurosciences*, 18(10):451–458, 1995. 91

[50] Stanislas Dehaene. *How we learn: Why brains learn better than any machine... for now*. Penguin, 2021. 3

[51] Emily Denton and Rob Fergus. Stochastic video generation with a learned prior. In *International conference on machine learning*, pages 1174–1183. PMLR, 2018. 51

[52] James J DiCarlo, Davide Zoccolan, and Nicole C Rust. How does the brain solve visual object recognition? *Neuron*, 73(3):415–434, 2012. 93

[53] Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014. 51

[54] Dawei W Dong and Joseph J Atick. Statistics of natural time-varying images. *Network: Computation in Neural Systems*, 6(3):345, 1995. 8, 9

[55] Dawei W Dong and Joseph J Atick. Temporal decorrelation: a theory of lagged and nonlagged responses in the lateral geniculate nucleus. *Network: Computation in neural systems*, 6(2):159, 1995. 91

[56] Rodney J Douglas, Kevan AC Martin, and David Whitteridge. A canonical microcircuit for neocortex. *Neural computation*, 1(4):480–488, 1989. 76

[57] Chaitanya Ekanadham, Daniel Tranchina, and Eero P Simoncelli. Recovery of sparse translation-invariant signals with continuous basis pursuit. *IEEE transactions on signal processing*, 59(10):4735–4744, 2011. 28

[58] Yasmine El-Shamayleh, Romesh D Kumbhani, Neel T Dhruv, and J Anthony Movshon. Visual response properties of v1 neurons projecting to v2 in macaque. *Journal of Neuroscience*, 33(42):16594–16605, 2013. 100

[59] Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 1:1, 2021. 43

[60] Peter Elias. Predictive coding–i. *IRE transactions on information theory*, 1(1):16–24, 1955. 19

[61] Jeffrey L Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990. 4

[62] Christina Enroth-Cugell and John G Robson. The contrast sensitivity of retinal ganglion cells of the cat. *The Journal of physiology*, 187(3):517–552, 1966. 76

[63] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6202–6211, 2019. 53

[64] Daniel J Felleman and David C Van Essen. Distributed hierarchical processing in the primate cerebral cortex. *Cerebral cortex (New York, NY: 1991)*, 1(1):1–47, 1991. 93

[65] David J Field. Relations between the statistics of natural images and the response properties of cortical cells. *Josa a*, 4(12):2379–2394, 1987. 10

[66] Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. *Advances in Neural Information Processing Systems*, 29, 2016. 51

[67] David J Fleet and Allan D Jepson. Computation of component image velocity from local phase information. *International journal of computer vision*, 5(1):77–104, 1990. 24, 36

[68] David J Fleet, Allan D Jepson, and Michael RM Jenkin. Phase-based disparity measurement. *CVGIP: Image understanding*, 53(2):198–210, 1991. 24

[69] Peter Földiák. Learning invariance from transformation sequences. *Neural Computation*, 3(2):194–200, 1991. 18, 89

[70] Jean Baptiste Joseph Fourier. *Théorie analytique de la chaleur*. Firmin Didot, 1822. 3

[71] Jeremy Freeman and Eero P Simoncelli. Metamers of the ventral stream. *Nature neuroscience*, 14(9):1195–1201, 2011. 103

[72] William T Freeman, Edward H Adelson, et al. The design and use of steerable filters. *IEEE Transactions on Pattern analysis and machine intelligence*, 13(9):891–906, 1991. 10, 31

[73] Karl Friston. The free-energy principle: a unified brain theory? *Nature reviews neuroscience*, 11(2):127–138, 2010. 88

[74] Kunihiko Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, 36(4):193–202, 1980. 77, 94

[75] Juan A Gallego, Matthew G Perich, Stephanie N Naufel, Christian Ethier, Sara A Solla, and Lee E Miller. Cortical population activity within a preserved neural manifold underlies multiple motor behaviors. *Nature communications*, 9(1):4233, 2018. 90

[76] Elad Ganmor, Michael S Landy, and Eero P Simoncelli. Near-optimal integration of orientation information across saccades. *Journal of vision*, 15(16):8–8, 2015. 103

[77] Ricardo Gattass, Charles G Gross, and Julie H Sandell. Visual topography of v2 in the macaque. *Journal of Comparative Neurology*, 201(4):519–539, 1981. 93

[78] Alexander Genkin, David Lipshutz, Siavash Golkar, Tiberiu Tesileanu, and Dmitri Chklovskii. Biological learning of irreducible representations of commuting transformations. *Advances in Neural Information Processing Systems*, 2022. 89

[79] Mathieu Germain, Karol Gregor, Iain Murray, and Hugo Larochelle. Made: Masked autoencoder for distribution estimation. In *International conference on machine learning*, pages 881–889. PMLR, 2015. 52

[80] James J Gibson. *The perception of the visual world.* Houghton Mifflin, 1950. 12

[81] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in Neural Information Processing Systems*, 27, 2014. 52

[82] Ross Goroshin, Michael F Mathieu, and Yann LeCun. Learning to linearize under uncertainty. In *Advances in Neural Information Processing Systems*, pages 1234–1242, 2015. 45

[83] Goesta H Granlund. In search of a general picture processing operator. *Computer Graphics and Image Processing*, 8(2):155–173, 1978. 94

[84] Robert M Gray. A survey of linear predictive coding. *Foundations and Trends® in Signal Processing*, 3(3):153–202, 2010. 20

[85] Florentin Guth, John Zarka, and Stephane Mallat. Phase collapse in neural networks. *International Conference on Learning Representations (ICLR)*, 2022. 28

[86] Brian C Hall. Lie groups, lie algebras, and representations. In *Quantum Theory for Mathematicians*, pages 333–366. Springer, 2013. 31

[87] Uri Hasson, Eunice Yang, Ignacio Vallines, David J Heeger, and Nava Rubin. A hierarchy of temporal receptive windows in human cortex. *Journal of neuroscience*, 28 (10):2539–2550, 2008. 94

[88] James H Hedges, Yevgeniya Gartshteyn, Adam Kohn, Nicole C Rust, Michael N Shadlen, William T Newsome, and J Anthony Movshon. Dissociation of neuronal and psychophysical responses to local and global motion. *Current Biology*, 21(23): 2023–2028, 2011. 103

[89] David J Heeger. Normalization of cell responses in cat striate cortex. *Visual neuroscience*, 9(2):181–197, 1992. 82, 84

[90] David J Heeger. Theory of cortical function. *Proceedings of the National Academy of Sciences*, 114(8):1773–1782, 2017. 89

[91] Jay Hegdé and David C Van Essen. Selectivity for complex shapes in primate visual area v2. *The Journal of Neuroscience*, 20(5):RC61, 2000. 76

[92] Olivier J Hénaff, Robbe LT Goris, and Eero P Simoncelli. Perceptual straightening of natural videos. *Nature neuroscience*, 22(6):984, 2019. 14

[93] Olivier J Hénaff, Yoon Bai, Julie A Charlton, Ian Nauhaus, Eero P Simoncelli, and Robbe LT Goris. Primary visual cortex straightens natural video trajectories. *Nature communications*, 12(1):5982, 2021. 15, 89, 90

[94] Janis K Hesse and Doris Y Tsao. The macaque face patch system: a turtle's underbelly for the brain. *Nature Reviews Neuroscience*, 21(12):695–716, 2020. 77

[95] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 56

[96] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646, 2022. 56

[97] Ha Hong, Daniel LK Yamins, Najib J Majaj, and James J DiCarlo. Explicit information for category-orthogonal object properties increases along the ventral stream. *Nature neuroscience*, 19(4):613–622, 2016. 79

[98] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012. 31

[99] Haruo Hosoya and Aapo Hyvärinen. A hierarchical statistical model of natural images explains tuning properties in v2. *Journal of Neuroscience*, 35(29):10412–10428, 2015. 94

[100] Alston S Householder. Unitary triangularization of a nonsymmetric matrix. *Journal of the ACM (JACM)*, 5(4):339–342, 1958. 42

[101] David H Hubel and Torsten N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1): 106, 1962. 76

[102] Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4), 2005. 53

[103] Aapo Hyvärinen and Patrik Hoyer. Emergence of phase-and shift-invariant features by decomposition of natural images into independent feature subspaces. *Neural computation*, 12(7):1705–1720, 2000. 85

[104] Edwin T Jaynes. How does the brain do plausible reasoning? technical report 421. *Reprinted in: Maximum-Entropy and Bayesian Methods in Science and Engineering. G.J. Erickson and C.R. Smitt (eds)*, 1:1–23, 1957. 6

[105] Linxing Preston Jiang and Rajesh PN Rao. Predictive coding theories of cortical function. *arXiv preprint arXiv:2112.10048*, 2021. 89, 94

[106] Linxing Preston Jiang and Rajesh PN Rao. Dynamic predictive coding: A new model of hierarchical sequence learning and prediction in the cortex. *bioRxiv*, pages 2022–06, 2022. 89, 94

[107] Elizabeth N Johnson, Michael J Hawken, and Robert Shapley. The spatial transforma-

tion of color in the primary visual cortex of the macaque monkey. *Nature neuroscience*, 4(4):409–416, 2001. 87

[108] Michael I Jordan. Serial order: A parallel distributed processing approach. In *Tech. Rep. No. 8604*. San Diego: University of California, Institute for Cognitive Science, 1986. 4

[109] Zahra Kadkhodaie and Eero Simoncelli. Stochastic solutions for linear inverse problems using the prior implicit in a denoiser. *Advances in Neural Information Processing Systems*, 34:13242–13254, 2021. 56, 60

[110] Zahra Kadkhodaie, Florentin Guth, Stéphane Mallat, and Eero P Simoncelli. Learning multi-scale local conditional probability models of images. *Int'l Conf on Learning Representations (ICLR)*, 2023. 62

[111] Zahra Kadkhodaie, Florentin Guth, Eero P Simoncelli, and Stéphane Mallat. Generalization in diffusion models arises from geometry-adaptive harmonic representation. *Int'l Conf on Learning Representations (ICLR)*, 2024. 56, 75

[112] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 1960. 3, 50

[113] Kohitij Kar and James J DiCarlo. Fast recurrent processing via ventrolateral prefrontal cortex is needed by the primate ventral stream for robust core visual object recognition. *Neuron*, 109(1):164–176, 2021. 101

[114] Kohitij Kar, Jonas Kubilius, Kailyn Schmidt, Elias B Issa, and James J DiCarlo. Evidence that recurrent circuits are critical to the ventral stream's execution of core object recognition behavior. *Nature neuroscience*, 22(6):974–983, 2019. 101

[115] Philip J Kellman and Elizabeth S Spelke. Perception of partly occluded objects in infancy. *Cognitive psychology*, 15(4):483–524, 1983. 102

[116] Ann Kennedy, Greg Wayne, Patrick Kaifosh, Karina Alviña, LF Abbott, and Nathaniel B Sawtell. A temporal basis for predicting the sensory consequences of motor commands in an electric fish. *Nature neuroscience*, 17(3):416–422, 2014. 4

[117] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *ICLR*, 2015. 110

[118] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *International Conference on Learning Representations*, 2014. 51

[119] David C Knill and Whitman Richards. *Perception as Bayesian inference.* Cambridge University Press, 1996. 6

[120] Dmitry Kobak, Wieland Brendel, Christos Constantinidis, Claudia E Feierstein, Adam Kepecs, Zachary F Mainen, Xue-Lian Qi, Ranulfo Romo, Naoshige Uchida, and Christian K Machens. Demixed principal component analysis of neural population data. *elife*, 5:e10989, 2016. 90

[121] Andrey Kolmogoroff. Interpolation und extrapolation von stationaren zufalligen folgen. *Izvestiya Rossiiskoi Akademii Nauk. Seriya Matematicheskaya*, 5(1):3–14, 1941. 3

[122] Imre Risi Kondor. *Group theoretical methods in machine learning.* Columbia University, 2008. 44

[123] Zoe Kourtzi and Nancy Kanwisher. Activation in human mt/mst by static images with implied motion. *Journal of cognitive neuroscience*, 12(1):48–55, 2000. 101

[124] Bart Krekelberg, Sabine Dannenberg, Klaus-Peter Hoffmann, Frank Bremmer, and John Ross. Neural correlates of implied motion. *Nature*, 424(6949):674–677, 2003. 101

[125] Rahul G Krishnan, Uri Shalit, and David Sontag. Deep kalman filters. *arXiv preprint arXiv:1511.05121*, 2015. 51

[126] Stephen W Kuffler. Discharge patterns and functional organization of mammalian retina. *Journal of neurophysiology*, 16(1):37–68, 1953. 76

[127] Pierre Simon Laplace. *Théorie analytique des probabilités*. Paris, 1812. 3

[128] Yann LeCun. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. *Open Review*, 62(1), 2022. 3

[129] Ann B Lee, David Mumford, and Jinggang Huang. Occlusion models for natural images: A statistical study of a scale-invariant dead leaves model. *International Journal of Computer Vision*, 41:35–59, 2001. 63

[130] Jörn-Philipp Lies, Ralf M Häfner, and Matthias Bethge. Slowness and sparseness have diverging effects on complex cell learning. *PLoS computational biology*, 10(3):e1003468, 2014. 45

[131] William Lotter, Gabriel Kreiman, and David Cox. Deep predictive coding networks for video prediction and unsupervised learning. *International Conference on Learning Representations*, 2017. 14, 53

[132] William Lotter, Gabriel Kreiman, and David Cox. A neural network trained for prediction mimics diverse features of biological neurons and perception. *Nature machine intelligence*, 2(4):210–219, 2020. 14

[133] Bethany Lusch, J Nathan Kutz, and Steven L Brunton. Deep learning for universal linear embeddings of nonlinear dynamics. *Nature communications*, 9(1):1–10, 2018. 21

[134] David JC MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003. 6

[135] George W Mackey. Harmonic analysis as the exploitation of symmetry–a historical survey. *Bulletin (New Series) of the American Mathematical Society*, 3(1. P1):543–698, 1980. 31

[136] John Makhoul. Linear prediction: A tutorial review. *Proceedings of the IEEE*, 63(4): 561–580, 1975. 20

[137] Stéphane Mallat. Group invariant scattering. *Communications on Pure and Applied Mathematics*, 65(10):1331–1398, 2012. 97

[138] Valerio Mante, Robert A Frazor, Vincent Bonin, Wilson S Geisler, and Matteo Carandini. Independence of luminance and contrast in natural scenes and in the early visual system. *Nature neuroscience*, 8(12):1690–1697, 2005. 91

[139] Valerio Mante, Vincent Bonin, and Matteo Carandini. Functional mechanisms shaping lateral geniculate responses to artificial and natural stimuli. *Neuron*, 58(4):625–638, 2008. 91

[140] Andrey A Markov. An example of statistical investigation of the text Eugene Onegin concerning the connection of samples in chains. (in Russian). *Bulletin of the Imperial Academy of Sciences of St. Petersburg 7(3):153–162.*, 1913. 3

[141] Harold Markowitz. Portfolio selection. *The Journal of Finance*, 7:77—-91, 1952. 4

[142] Georges Matheron. *Random sets and integral geometry.* John Wiley and Sons, 1975. 63

[143] Michael Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. *ICLR*, 2016. 38, 52, 108

[144] Roland Memisevic and Geoffrey E Hinton. Learning to represent spatial transformations with factored higher-order boltzmann machines. *Neural computation*, 22(6): 1473–1492, 2010. 45

[145] Igor Mezić. Spectral properties of dynamical systems, model reduction and decompositions. *Nonlinear Dynamics*, 41(1):309–325, 2005. 21

[146] Koichi Miyasawa et al. An empirical bayes estimator of the mean of a normal population. *Bull. Inst. Internat. Statist*, 38(181-188):1–2, 1961. 57

[147] Sreyas Mohan, Zahra Kadkhodaie, Eero P Simoncelli, and Carlos Fernandez-Granda. Robust and interpretable blind image denoising via bias-free convolutional neural networks. *Int'l Conf on Learning Representations (ICLR)*, 2020. 67, 70

[148] J Anthony Movshon, Edward H Adelson, Martin S Gizzi, and William T Newsome. The analysis of moving visual patterns. *Pattern Recognition Mechanisms, ed. C. Chagas, R. Gattass, C.Gross (Pontificiae Academiae Scientiarum Scripta Varia 54)*, pages 117–151, 1985. 76

[149] David Mumford. On the computational architecture of the neocortex: Ii the role of cortico-cortical loops. *Biological cybernetics*, 66(3):241–251, 1992. 88

[150] Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability. *International Conference on Machine Learning*, 2023. 44

[151] Lionel G Nowak and Jean Bullier. The timing of information transfer in the visual system. *Cerebral Cortex*, 12, 1997. 100

[152] Gouki Okazawa, Christina E Hatch, Allan Mancoo, Christian K Machens, and Roozbeh

Kiani. Representational geometry of perceptual decisions in the monkey parietal cortex. *Cell*, 184(14):3748–3761, 2021. 89

[153] Timothy D Oleskiw, Justin D Lieber, Eero P Simoncelli, and J Anthony Movshon. Foundations of visual form selectivity for neurons in macaque v1 and v2. *bioRxiv*, 2024. 99

[154] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016. 53

[155] Alan V Oppenheim and Jae S Lim. The importance of phase in signals. *Proceedings of the IEEE*, 69(5):529–541, 1981. 9

[156] Sergiu Oprea, Pablo Martinez-Gonzalez, Alberto Garcia-Garcia, John Alejandro Castro-Vargas, Sergio Orts-Escolano, Jose Garcia-Rodriguez, and Antonis Argyros. A review on deep learning techniques for video prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 49

[157] Stephanie E Palmer, Olivier Marre, Michael J Berry, and William Bialek. Predictive information in a sensory population. *Proceedings of the National Academy of Sciences*, 112(22):6908–6913, 2015. 14, 91

[158] N Parthasarathy, O J Hénaff, and E P Simoncelli. Layerwise complexity-matched learning yields an improved model of cortical area V2. *Trans. Machine Learning Research*, Jun 2024. 97

[159] W Pitts and WS McCulloch. How we know universals: the perception of auditory and visual forms. *Bulletin of Mathematical Biophysics, Vol. 9*, 1947. 78

[160] Tomaso A Poggio and Fabio Anselmi. *Visual cortex and deep networks: learning invariant representations*. MIT press, 2016. 79

[161] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alexander Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv:1704.00675*, 2017. 109

[162] Javier Portilla and Eero P Simoncelli. A parametric texture model based on joint statistics of complex wavelet coefficients. *International journal of computer vision*, 40 (1):49–70, 2000. 10, 24, 33, 99, 108

[163] Javier Portilla, Vasily Strela, Martin J Wainwright, and Eero P Simoncelli. Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE Transactions on Image processing*, 12(11):1338–1351, 2003. 11

[164] Vilayanur S Ramachandran and Stuart M Anstis. Illusory displacement of equiluminous kinetic edges. *Perception*, 19(5):611–616, 1990. 101

[165] MarcAurelio Ranzato, Arthur Szlam, Joan Bruna, Michael Mathieu, Ronan Collobert, and Sumit Chopra. Video (language) modeling: a baseline for generative models of natural videos. *arXiv preprint arXiv:1412.6604*, 2014. 52

[166] Rajesh Rao and Daniel Ruderman. Learning lie groups for invariant visual perception. *Advances in Neural Information Processing Systems*, 11, 1998. 44

[167] Rajesh PN Rao. An optimal estimation approach to visual perception and learning. *Vision research*, 39(11):1963–1989, 1999. 14

[168] Rajesh PN Rao and Dana H Ballard. Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature neuroscience*, 2(1):79–87, 1999. 14, 88

[169] Martin Raphan and Eero P Simoncelli. Least squares estimation without priors or supervision. *Neural computation*, 23(2):374–420, 2011. 54

[170] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International conference on machine learning*, pages 1530–1538. PMLR, 2015. 51

[171] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic back-propagation and approximate inference in deep generative models. In *International conference on machine learning*, pages 1278–1286. PMLR, 2014. 51

[172] Lewis Fry Richardson. Atmospheric diffusion shown on a distance-neighbour graph. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 110(756):709–737, 1926. 3

[173] Maximilian Riesenhuber and Tomaso Poggio. Hierarchical models of object recognition in cortex. *Nature neuroscience*, 2(11):1019–1025, 1999. 77

[174] Dario L Ringach. Spatial structure and symmetry of simple-cell receptive fields in macaque primary visual cortex. *Journal of neurophysiology*, 88(1):455–463, 2002. 91

[175] Herbert E Robbins. An empirical bayes approach to statistics. In *Proceedings of the Third Berkeley Symposium on Probability and Statistics*, pages 157–163, 1956. 57

[176] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention*, pages 234–241, 2015. 38, 53, 62, 108, 111

[177] Ruth Rosenholtz. Capabilities and limitations of peripheral vision. *Annual review of vision science*, 2(1):437–457, 2016. 103

[178] Sam Roweis and Zoubin Ghahramani. A unifying review of linear gaussian models. *Neural computation*, 11(2):305–345, 1999. 50

[179] DL Ruderman. The statistics of narual images. *Network: computation in Neural Systems*, 5(517-548):65–72, 1994. 11

[180] Nicole C Rust and James J DiCarlo. Selectivity and tolerance ("invariance") both increase as visual information propagates from cortical area v4 to it. *Journal of Neuroscience*, 30(39):12978–12995, 2010. 93

[181] Nicole C Rust and Stephanie E Palmer. Remembering the past to see the future. *Annual review of vision science*, 7:349–365, 2021. 93

[182] Nicole C Rust, Odelia Schwartz, J Anthony Movshon, and Eero P Simoncelli. Spatiotemporal elements of macaque v1 receptive fields. *Neuron*, 46(6):945–956, 2005. 91

[183] ML Sabin and R Gray. Product code vector quantizers for waveform and voice coding. *IEEE transactions on acoustics, speech, and signal processing*, 32(3):474–488, 1984. 24

[184] Sophia Sanborn, Christian Shewmake, Bruno Olshausen, and Christopher Hillar. Bispectral neural networks. *arXiv preprint arXiv:2209.03416*, 2022. 45

[185] David M Schneider, Janani Sundararajan, and Richard Mooney. A cortical filter that learns to suppress the acoustic consequences of movement. *Nature*, 561(7723):391, 2018. 4

[186] Martin Schrimpf, Jonas Kubilius, Ha Hong, Najib J Majaj, Rishi Rajalingham, Elias B Issa, Kohitij Kar, Pouya Bashivan, Jonathan Prescott-Roy, Franziska Geiger, et al. Brain-score: Which artificial neural network for object recognition is most brain-like? *BioRxiv*, page 407007, 2018. 77

[187] Odelia Schwartz and Eero P Simoncelli. Natural signal statistics and sensory gain control. *Nature neuroscience*, 4(8):819, 2001. 11

[188] Claude E Shannon. Prediction and entropy of printed english. *Bell system technical journal*, 30(1):50–64, 1951. 3

[189] Dev Yashpal Sheth, Sreyas Mohan, Joshua L Vincent, Ramon Manzorro, Peter A Crozier, Mitesh M Khapra, Eero P Simoncelli, and Carlos Fernandez-Granda. Unsupervised deep video denoising. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1759–1768, 2021. 71

[190] E P Simoncelli, W T Freeman, E H Adelson, and D J Heeger. Shiftable multi-scale transforms. *IEEE Trans Information Theory*, 38(2):587–607, Mar 1992. doi: 10.1109/18.119725. Special Issue on Wavelets. 10

[191] Eero P. Simoncelli. *Distributed representation and analysis of visual motion.* PhD thesis, Massachusetts Institute of Technology, 1993. 12

[192] Eero P Simoncelli. Optimal estimation in sensory systems. *The Cognitive Neurosciences, IV*, pages 525–535, 2009. 6

[193] Eero P Simoncelli and Edward H Adelson. Noise removal via bayesian wavelet coring. In *Proceedings of 3rd IEEE international conference on image processing*, volume 1, pages 379–382. IEEE, 1996. 10

[194] Eero P Simoncelli and William T Freeman. The steerable pyramid: A flexible architecture for multi-scale derivative computation. *Proceedings., International Conference on Image Processing*, 3:444–447, 1995. 38, 108

[195] Eero P Simoncelli and David J Heeger. A model of neuronal responses in visual area mt. *Vision research*, 38(5):743–761, 1998. 12, 84, 95, 99

[196] Eero P Simoncelli and Bruno A Olshausen. Natural image statistics and neural representation. *Annual review of neuroscience*, 24(1):1193–1216, 2001. 5

[197] Yosef Singer, Yayoi Teramoto, Ben DB Willmore, Jan WH Schnupp, Andrew J King, and Nicol S Harper. Sensory cortex is optimized for prediction of future input. *Elife*, 7:e31557, 2018. 22

[198] Yosef Singer, Luke Taylor, Ben DB Willmore, Andrew J King, and Nicol S Harper. Hierarchical temporal prediction captures motion processing along the visual pathway. *Elife*, 12:e52599, 2023. 94

[199] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019. 56

[200] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *International Conference on Learning Representations (ICLR)*, 2021. 56

[201] Elizabeth S Spelke. Principles of object perception. *Cognitive science*, 14(1):29–56, 1990. 102

[202] Mandyam Veerambudi Srinivasan, Simon Barry Laughlin, and Andreas Dubs. Predictive coding: a fresh view of inhibition in the retina. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 216(1205):427–459, 1982. 91

[203] Peter Sterling. *What is health?: Allostasis and the evolution of human design*. MIT Press, 2020. 2

[204] Peter Sterling and Simon Laughlin. *Principles of neural design*. MIT press, 2015. 5

[205] Jordan W Suchow and George A Alvarez. Motion silences awareness of visual change. *Current Biology*, 21(2):140–143, 2011. 102

[206] Simon Thorpe, Denis Fize, and Catherine Marlot. Speed of processing in the human visual system. *nature*, 381(6582):520–522, 1996. 100

[207] Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method. In *Proc. of the 37-th Annual Allerton Conference on Communication, Control and Computing*, pages 368–377, 1999. 13

[208] Leslie G Ungerleider and Mortimer Mishkin. Two cortical visual systems. In *Analysis of visual behavior*. MIT Press Cambridge, MA, 1982. 101

[209] Aäron Van Den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In *International conference on machine learning*, pages 1747–1756. PMLR, 2016. 52

[210] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in Neural Information Processing Systems*, 30, 2017. 52

[211] J Hans van Hateren and Dan L Ruderman. Independent component analysis of natural image sequences yields spatio-temporal filters similar to simple cells in primary visual cortex. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 265 (1412):2315–2320, 1998. 109

[212] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017. 41

[213] Hermann Von Helmholtz. *Handbuch der physiologischen Optik*, volume 9. Voss, 1867. 5

[214] Neal Wadhwa, Michael Rubinstein, Frédo Durand, and William T Freeman. Phase-based video motion processing. *ACM Transactions on Graphics*, 32(4):1–10, 2013. 25

[215] Martin J Wainwright and Eero Simoncelli. Scale mixtures of gaussians and the statistics of natural images. *Advances in Neural Information Processing Systems*, 12, 1999. 11

[216] Martin J Wainwright, Michael I Jordan, et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2): 1–305, 2008. 51

[217] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. *arXiv preprint arXiv:1808.06601*, 2018. 52

[218] Zhou Wang and Eero Simoncelli. Local phase coherence and the perception of blur. *Advances in Neural Information Processing Systems*, 16, 2003. 24

[219] Eiji Watanabe, Akiyoshi Kitaoka, Kiwako Sakamoto, Masaki Yasugi, and Kenta Tanaka. Illusory motion reproduced by deep neural networks trained for prediction. *Frontiers in psychology*, 9:345, 2018. 102

[220] SG Waxman and Michael VL Bennett. Relative conduction velocities of small myelinated and non-myelinated fibres in the central nervous system. *Nature New Biology*, 238(85):217–219, 1972. 100

[221] Yair Weiss, Eero P Simoncelli, and Edward H Adelson. Motion illusions as optimal percepts. *Nature neuroscience*, 5(6):598–604, 2002. 12

[222] Thomas Wiegand, Gary J Sullivan, Gisle Bjontegaard, and Ajay Luthra. Overview of the H.264/AVC video coding standard. *IEEE Transactions on circuits and systems for video technology*, 13(7):560–576, 2003. 4, 19

[223] Norbert Wiener. The extrapolation, interpolation, and smoothing of stationary time series. *Report of the Services 19, Research Project DIC-6037*, 1942. 3, 50, 56

[224] Jonathan Winawer, Alexander C Huk, and Lera Boroditsky. A motion aftereffect from still photographs depicting motion. *Psychological Science*, 19(3):276–283, 2008. 101

[225] Laurenz Wiskott and Terrence J Sejnowski. Slow feature analysis: Unsupervised learning of invariances. *Neural computation*, 14(4):715–770, 2002. 18, 21, 92

[226] Andrew P Witkin. Scale-space filtering. In *Proceedings of IJCAI*, pages 1019–1021. Karlsruhe, 1983. 59

[227] Daniel M Wolpert, Zoubin Ghahramani, and Michael I Jordan. An internal model for sensorimotor integration. *Science*, 269(5232):1880–1882, 1995. 4

[228] Hao-Yu Wu, Michael Rubinstein, Eugene Shih, John Guttag, Frédo Durand, and William Freeman. Eulerian video magnification for revealing subtle changes in the world. *ACM transactions on graphics (TOG)*, 31(4):1–8, 2012. 25

[229] Robert H Wurtz. Neuronal mechanisms of visual stability. *Vision research*, 48(20): 2070–2089, 2008. 93

[230] Tianfan Xue, Jiajun Wu, Katherine Bouman, and Bill Freeman. Visual dynamics: Probabilistic future frame synthesis via cross convolutional networks. *Advances in Neural Information Processing Systems*, 29, 2016. 53

[231] Daniel LK Yamins, Ha Hong, Charles F Cadieu, Ethan A Solomon, Darren Seibert, and James J DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the national academy of sciences*, 111(23):8619–8624, 2014. 77

[232] George Udny Yule. On a method of investigating periodicities disturbed series, with special reference to wolfer's sunspot numbers. *Philosophical Transactions of the Royal*

Society of London. Series A, Containing Papers of a Mathematical or Physical Character, 226(636-646):267–298, 1927. 20

[233] Andrew D Zaharia, Robbe LT Goris, J Anthony Movshon, and Eero P Simoncelli. Compound stimuli reveal the structure of visual motion selectivity in macaque mt neurons. *Eneuro*, 6(6), 2019. 12, 99

[234] Christoph Zetzsche, Gerhard Krieger, and Bernhard Wegmann. The atoms of vision: Cartesian or polar? *JOSA A*, 16(7):1554–1565, 1999. 24

[235] Sixin Zhang and Stéphane Mallat. Maximum entropy models from phase harmonic covariances. *Applied and Computational Harmonic Analysis*, 53:199–230, 2021. 33

[236] Yuan Zhao and Il Memming Park. Variational latent gaussian process for recovering single-trial dynamics from population spike trains. *Neural computation*, 29(5):1293–1316, 2017. 90

[237] Shan Zhu and Kai-Kuang Ma. A new diamond search algorithm for fast block-matching motion estimation. *IEEE transactions on Image Processing*, 9(2):287–290, 2000. 107

[238] Corey M Ziemba and Eero P Simoncelli. Opposing effects of selectivity and invariance in peripheral vision. *Nature communications*, 12(1):4597, 2021. 103

[239] Corey M Ziemba, Jeremy Freeman, J Anthony Movshon, and Eero P Simoncelli. Selectivity and tolerance for visual texture in macaque v2. *Proceedings of the National Academy of Sciences*, 113(22):E3140–E3149, 2016. 99

[240] Corey M Ziemba, Richard K Perez, Julia Pai, Jenna G Kelly, Luke E Hallum, Christopher Shooner, and J Anthony Movshon. Laminar differences in responses to naturalistic texture in macaque v1 and v2. *Journal of Neuroscience*, 39(49):9748–9756, 2019. 101