12 Efficient Coding of Time-Varying Signals Using a Spiking Population Code

Michael S. Lewicki

Introduction

Representing time varying signals is a fundamental problem in processing any signal orignating from the natural environment, but there is no natural way to encode such signals, and traditional methods each have their limitations. A common method of describing temporal signal is to divide it into a sequence of blocks. The data within each block is then fit or decomposed with standard basis such as a Fourier or wavelet. Blocking the data has the limitation that the components of the bases are arbitrarily aligned with respect to structure in the time series. Figure 12.1 shows a short segment of speech data and the boundaries of the blocks. Although the structure in the signal is largely periodic, each large oscillation appears in a different position within the blocks and is sometimes split across blocks. This problem is particularly acute for acoustic events with sharp onset, such as plosives in speech. It also presents difficulties for encoding the signal efficiently, because there is no compact way to describe phase-dependent structure. This can be somewhat circumvented by techniques such as windowing or averaging sliding blocks, but it would be more desirable if the representation were phase or shift invariant [15].

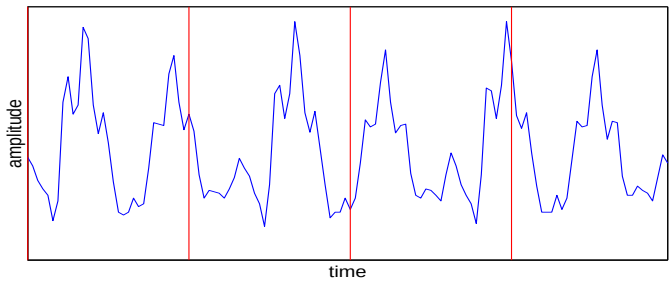

Figure 12.1. Blocking results in arbitrary phase alignment of the underlying structure.

An example of a shift invariant representation is a filter bank, shown in fig-

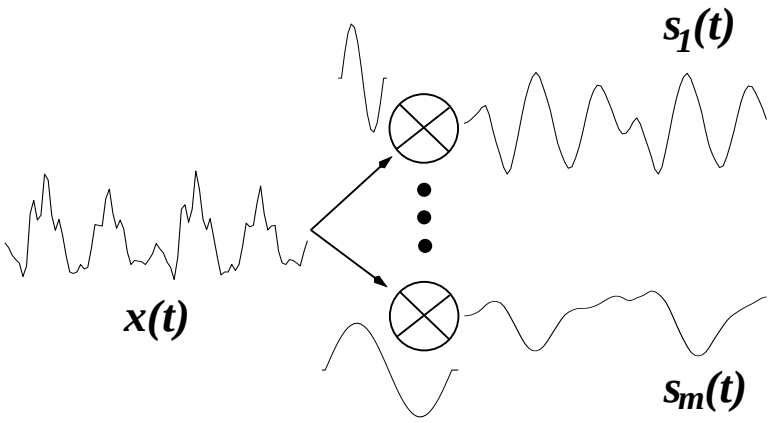

Figure 12.2. A filter bank representation of time-varying signals. The time-varying signal is convolved with different filters (represented by the $\otimes$ symbol), resulting in m different output signals. This representation is shift-invariant, but does not compactly represent structure in the original signal.

ure 12.2. Each unit convolves the input signal to produce a time-varying output signal. This type of representation is implicit in many models of neural processing where the input and output signals are represented by average firing rates, and the convolutions performed by the different units correspond to the spatio-temporal receptive fields of different neurons.

The limitation of this representation is that it doesn't capture any of the temporal structure of the signal, it simply converts one time varying signal into many. If the input is a *set* of time-varying signals, the situation is somewhat different, because then one can attempt to find the set of transformations or filters that minimize the statistical dependencies among the outputs. This is exactly the goal of independent component analysis [9, 4] and approaches that seek factorial codes [3, 1, 17, 19]. Although some methods of been developed to make use of temporal structure for improving the statistical independence of the output signals [20, 2], a continuous output signal implies that the representation is not efficiently encoding the *temporal* structure in the input signal.

The Model

Our goal is to develop a model that will efficiently represent temporal structure in a time-varying input signal. This is accomplished by modeling that the signal by small set of *kernel* functions that can be placed at arbitrary time points. Ultimately, we want to find the minimal set of functions and time points that fit the signal within a given noise level. We expect this type of model to work well for signals composed of events whose onset can occur at arbitrary temporal positions. Examples of these include, musical instruments sounds with sharp attack or plosive sounds in speech.

We assume time series $x(t)$ is modeled by

$$x(t) = \sum_i s_i \phi_{m[i]}(t - \tau_i) + \epsilon(t)\,, \tag{12.1}$$

where τ_i indicates the temporal position of the i^{th} kernel function, $\phi_{m[i]}$, which is scaled by s_i. The notation $m[i]$ represents an index function that specifies which of the M kernel functions is present at time τ_i. A single kernel function can occur at multiple times during the time series. Additive noise at time t is given by $\epsilon(t)$.

A more general way to express (12.1) is to assume that the kernel functions exist at all time points during the signal, and let the non-zero coefficients determine the positions of the kernel functions. In this case, the model can be expressed in convolutional form

$$x(t) = \sum_m \int s_m(\tau)\phi_m(t - \tau)d\tau + \epsilon(t) \tag{12.2}$$

$$= \sum_m s_m(t) * \phi_m(t) + \epsilon(t)\,, \tag{12.3}$$

where $s_m(\tau)$ is the coefficient at time τ for kernel function ϕ_m.

It is also helpful to express the model in matrix form using a discrete sampling of the continuous time series:

$$x = As + \epsilon\,. \tag{12.4}$$

The basis matrix, A, is defined by

$$A = [C(\phi_1)\, C(\phi_2)\, \cdots\, C(\phi_M)]\,, \tag{12.5}$$

where $C(a)$ is an N-by-N circulant matrix parameterized by the vector a. This matrix is constructed by replicating the kernel functions at each sample position

$$C(a) = \begin{bmatrix} a_0 & a_{N-1} & \ldots & a_2 & a_1 \\ a_1 & a_0 & \ldots & a_3 & a_2 \\ \ldots & & \ldots & & \ldots \\ a_{N-2} & a_{N-3} & \ldots & a_0 & a_{N-1} \\ a_{N-1} & a_{N-2} & \ldots & a_1 & a_0 \end{bmatrix} \tag{12.6}$$

The kernels are zero padded to be of length N. The length of each kernel is typically much less than the length of the signal, making A very sparse. This can be viewed as a special case of a Toeplitz matrix. Note that the size of A is MN-by-N, and is thus an example of an overcomplete basis, i.e. a basis with more basis functions than dimensions in the data space [21, 8, 18, 16].

A Probabilistic Formulation

The optimal coefficient values for a signal are found by maximizing the posterior distribution

$$\hat{s} = \arg\max_s P(s|x, A) = \arg\max_s P(x|A, s)P(s) \tag{12.7}$$

where $\hat{s}$ is the most probable representation of the signal. Note that omission of the normalizing constant $P(x|A)$ does not change the location of the maximum. This formulation of the problem offers the advantage that the model can fit more general types of distributions and naturally "denoises" the signal. Note that the mapping from x to $\hat{s}$ is *nonlinear* with non-zero additive noise and an overcomplete basis [7, 16]. Optimizing (12.7) essentially selects out the subset of basis functions that best account for the data.

To define a probabilistic model, we follow existing conventions for linear generative models with additive noise [6, 16]. We assume the noise, ϵ, to have a Gaussian distribution which yields a data likelihood for a given representation of

$$\log P(x|A, s) \propto -\frac{1}{2\sigma^2}(x - As)^2 \,. \tag{12.8}$$

The function $P(s)$ describes the a priori distribution of the coefficients. Under the assumption that $P(s)$ is sparse (highly peaked around zero), maximizing (12.7) results in very few nonzero coefficients. A compact representation of $\hat{s}$ is to describe the values of the non-zero coefficients and their temporal positions

$$P(s) = \prod_m P(u_m, \tau_m) = \prod_{m=1}^{M} \prod_{i=1}^{n_m} P(u_{m,i})P(\tau_{m,i}) \,, \tag{12.9}$$

where the prior for the non-zero coefficient values, $u_{m,i}$, is assumed to be Laplacian, and the prior for the temporal positions (or intervals), $\tau_{m,i}$, is assumed to be a gamma distribution.

Finding the Best Encoding

A difficult challenge presented by the proposed model is finding a computationally tractable method for fitting it to the data. The brute-force approach of generating the basis matrix A generates an intractable number basis functions for signals of any reasonable length, so we need to look for ways of reducing the computational cost of optimizating (12.7). We start with the gradient of the log posterior

$$\frac{\partial}{\partial s} \log P(s|A, x) \propto A^T(x - As) + z(s) \,, \tag{12.10}$$

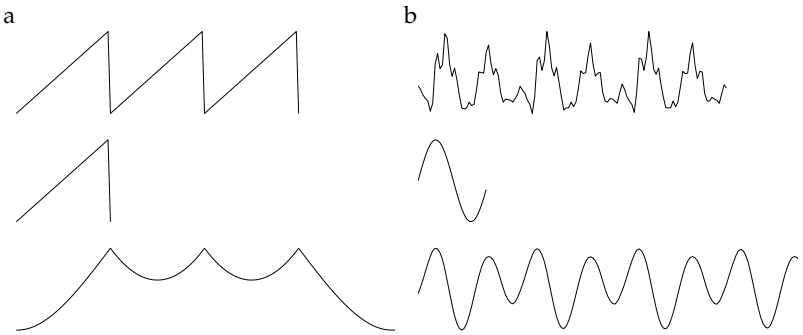

Figure 12.3. Convolution using the fast Fourier transform is an efficient way to select an initial solution for the temporal positions of the kernel functions. (**a**) The convolution of a sawtooth waveform with a sawtooth-shaped kernel function (middle). (**b**) Convolution of a speech segment with a single period sine-wave kernel function.

where $z(s) = (\log P(s))'$. A basic operation required is $v = A^T u$. We saw that $x = As$ can be computed efficiently using convolution (12.2). Because A^T is also block circulant

$$A^T = \begin{bmatrix} C(\phi_1') \\ \dots \\ C(\phi_M') \end{bmatrix} \tag{12.11}$$

where $\phi'(1:N) = \phi(N:-1:1)$. Thus, terms involving A^T can also be computed efficiently using convolution

$$v = A^T u = \begin{bmatrix} \phi_1(-t) * u(t) \\ \dots \\ \phi_M(-t) * u(t) \end{bmatrix} \tag{12.12}$$

Obtaining an initial representation

An alternative approach to optimizing (12.7) is to make use of the fact that if the kernel functions are short enough in length, direct multiplication is faster than convolution, and that, for this highly overcomplete basis, most of the coefficients will be zero after being fit to the data. The central problem in encoding the signal then is to determine which coefficients are non-zero, ideally finding a description of the time series with the minimal number of non-zero coefficients. This is equivalent to determining the best set of temporal positions for each of the kernel functions (12.1).

A crucial step in this approach is to obtain a good initial estimate of the coefficients. One way to do this is to consider the projection of the signal onto each of the basis functions, i.e. $A^T x$. This estimate will be exact (i.e. zero residual error) in the case of zero noise and A orthogonal. For the non-orthogonal, overcomplete case the solution will be approximate, but for certain choices of the basis matrix, an exact representation can still be obtained efficiently [10, 21].

Figure 12.3 shows examples of convolving two different kernel functions with data. One disadvantage with this initial solution is that the coefficient functions, $s_m(t)$, are not sparse. For example, even though the signal in figure 12.3a is composed of only three instances of the kernel function, the convolution is mostly non-zero.

A simple procedure for obtaining a better initial estimate of the most probable coefficients is to select the time locations of the maxima (or extrema) in the convolutions. These are positions where the kernel functions capture the greatest amount of signal structure and where the optimal coefficients are likely to be non-zero. Figure 12.4 shows the result of using this procedure to obtain an initial fit to a speech segment. This can generate a large number of kernel positions, but their number can be reduced further by selecting only those that contribute significantly, i.e. where the average power is greater than some fraction of the noise level. From these, a basis for the entire signal is constructed by replicating the kernel functions at the appropriate time positions.

Once an initial estimate and basis are formed, the most probable coefficient values are estimated using a modified conjugate gradient procedure. The size of the generated basis does not pose a problem for optimization, because it is has very few non-zero elements (the number of which is roughly constant per unit time). This arises because each column is non-zero only around the position of the kernel function, which is typically much shorter in duration than the data waveform. This structure affords the use of sparse matrix routines for all the key computations in the conjugate gradient routine. After the initial fit, there typically are a large number of basis functions that give a very small contribution. These can be pruned to yield, after refitting, a more probable representation that has significantly fewer coefficients.

Properties of the Representation

Figure 12.5 shows the results of fitting a segment of speech with a sine wave kernel set. This was composed of 64 kernel functions constructed using a single period of a sine function whose log frequencies were evenly distributed between 0 and Nyquist (4 kHz), which yielded kernel functions that were minimally correlated (they are not orthogonal because each has only one cycle and is zero elsewhere). The kernel function lengths varied between 5 and 64 samples. The plots show the positions of the non-zero coefficients superimposed on the waveform. The residual errors curves from the fitted waveforms are shown offset, below each waveform. The right axes indicate the kernel function number which increase with frequency. The dots show the starting position of the kernels with non-zero coefficients, with the dot size scaled according to the mean power contribution.

Figure 12.5a shows that the structure in the coefficients repeats for each oscillation in the waveform. Adding a delay leaves the relative temporal structure of the non-zero coefficients mostly unchanged (figure 12.5b). The small variations between the two sets of coefficients are due to variations in the fitting of the small-

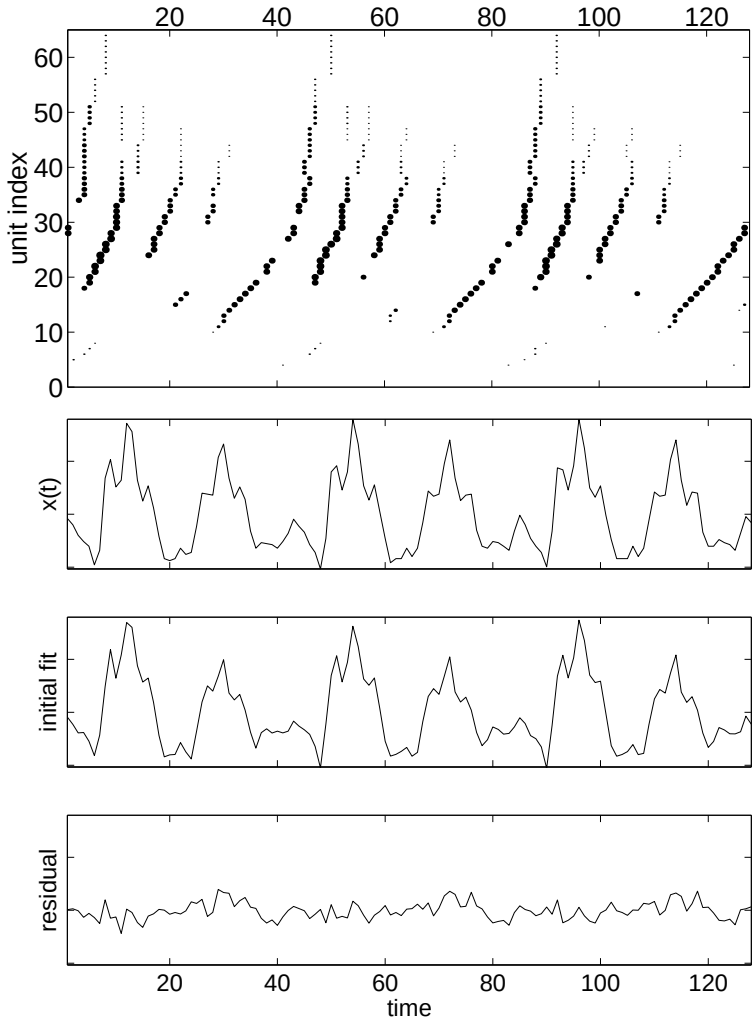

Figure 12.4. The initial fitting procedure of selecting kernel function positions at convolution peaks. The dots in the upper plot indicate positions of kernels (right axis) with size scaled by the mean power contribution. The original and initial reconstructed speech signal are plotted below, with the bottom plot showing the residual error (12 dB SNR). The residual error can be improved to 70dB SNR after optimizing the coefficient magnitudes (figure 12.5a).

magnitude coefficients. Representing the signal in figure 12.5b with a standard complete basis would result in a very different representation.

Fine-scale time frequency analysis

In contrast to Fourier or wavelet decompositions, this type of representation or decomposition can place the kernel functions at arbitrary time points. In the special case of sinusoid kernel functions, the decomposition performs a fine-scale time-frequency analysis, an example of which is illustrated in figure 12.6. The plot of the kernel function positions show how the representation picks up the high frequency structure near the beginning of the waveform. In a Fourier decomposi-

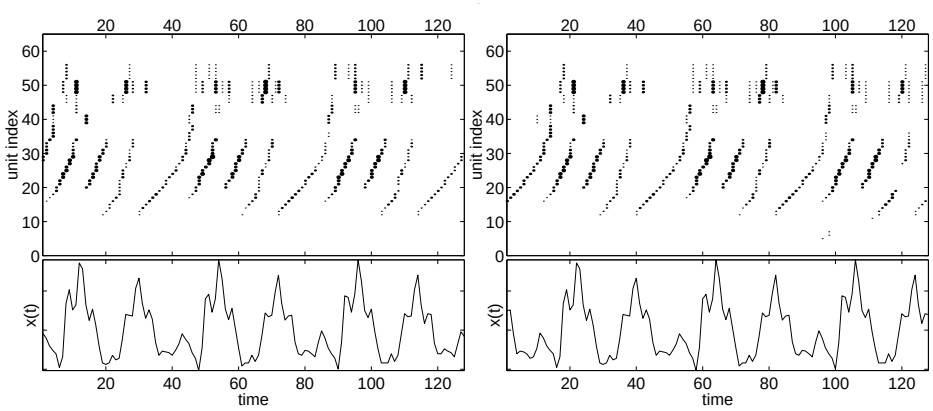

Figure 12.5. Fitting a shift-invariant model to a segment of speech, x(t). (a) Shows the fit to the original unshifted signal. The accuracy of the fit is 70 dB SNR. (b) The fit to a shifted version of the same signal.

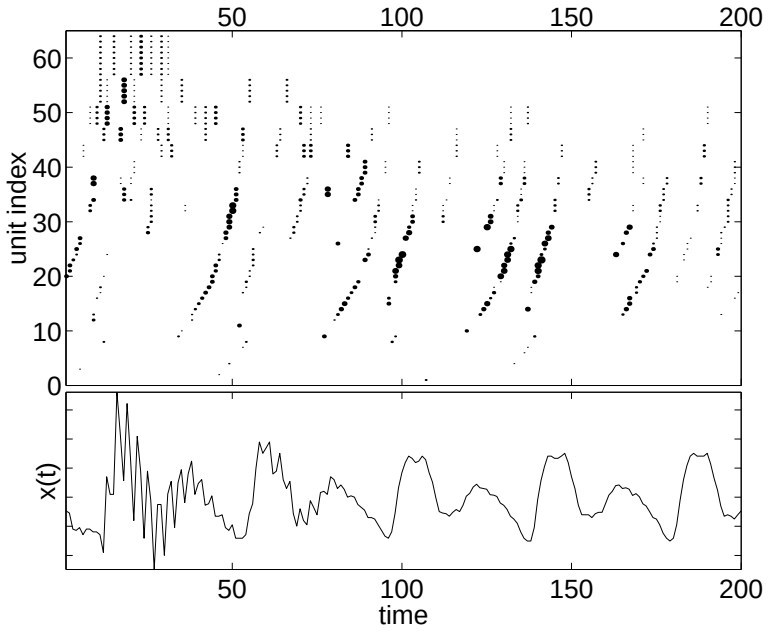

Figure 12.6. An example showing how, in the case of sinusoid kernel functions, the decomposition performs a fine-grained time-frequency analysis.

tion, the high frequency energy would only be localized to within the window. A wavelet decomposition would allow better temporal localization, but would still be limited to a set of discrete temporal positions.

Neural Implementations

The initial representation is implemented using a simple convolution plus threshold. How could coefficient magnitudes be coded with fixed amplitude action po-

tentials? We consider several models of biologically plausible models of spike coding [13]. Figure 12.7 shows several possibilities. In figure 12.7a, coefficient magnitude is encoded by the average firing rate. This is a classic model of neural coding of analogue values and can be implemented simply by making firing probability increase with increasing input. The disadvantage of this model is that temporal precision is lost.

Figure 12.7b shows a model that encodes the magnitude of the kernel functions using a distributed population code. In this model, several neurons with the same or similar convolution properties firing probabilistically in response to the input magnitude. The analog signal is transmitted using a population of spikes, but can be recovered at the post-synaptic neural simply by summation. In contrast to the average firing rate model, this model preserves temporal precision, but at the expense of addition units. This type of model offers on explanation for why overcomplete representations might be useful in neural coding.

The model in figure 12.7c encodes magnitude using the position relative to a common background oscillation that influences the firing times of neurons in the population. Neurons that receive strong input (magnitude) will fire earlier, thus encoding magnitude in terms of relative spike timing. Having a common background oscillation can serve both to establish a common reference for the population and to reduce the variability in the relative firing times.

Figure 12.7d shows a model that makes use of the same mechanisms as that in 12.7c, i.e. neurons receiving larger magnitude input will reach the spiking threshold faster, but does not use a common oscillation. Magnitude information is conveyed in the relative timing or synchrony of the spikes, and could be decoded by giving more weight to earlier action potentials. An assumption of this model is that random timing coincidences will be interpreted as noise. An advantage of this and the previous model is that no temporal precision is lost and redundancy in the unit response properties is not required, although that may still be advantageous for compensating for intrinsic noise in the population or limitations of spike timing precision.

Discussion

The model presented here can be viewed as an extension of the shiftable transforms of [21]. One difference is that here no constraints are placed on the kernel functions. Furthermore, this model accounts for additive noise, which yields automatic signal denoising and provides sensible criteria for selecting significant coefficients. An important unresolved issue is how well the algorithm works for increasingly non-orthogonal kernels.

Representing a time-varying signal in terms of a sparse set of kernel functions is exactly the model assumed in the approach of reverse correlation or stimulus reconstruction models [11, 12, 5], but the goals are converse, i.e. rather from going from a single spike train to an estimate of the stimulus, the model and algorithm described here go from a stimulus to its optimal representation in a *population*

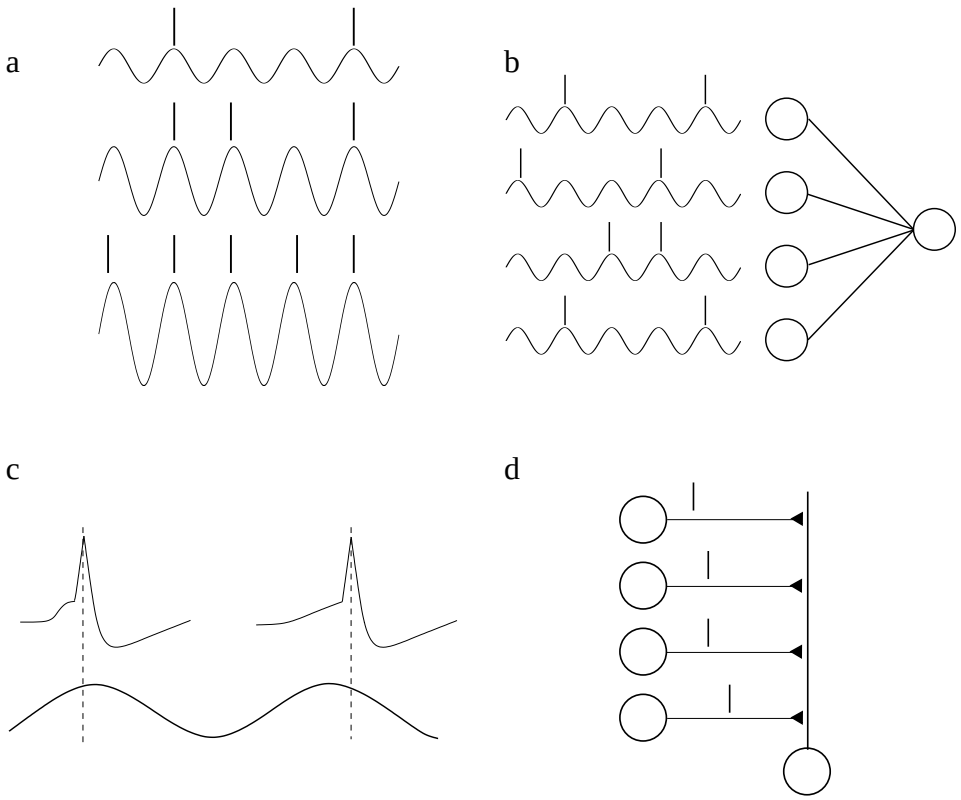

Figure 12.7. (a) Spike frequency model. Convolution magnitude is encoded by the average firing rate, with some loss of temporal precision. (b) Population spike model. Convolution magnitude is encoded by a population of neurons with similar kernels. If each unit fires probabilistically, the magnitude of the convolution is recovered in the post-synaptic sum. (c) Phase model. Convolution magnitude is encoded by the timing of the action potential relative to a common oscillation using the property that larger magnitude inputs will reach the spiking threshold faster. (d) Relative timing model. Magnitude is again coded by relative spike timing, but without a common oscillation.

of spike trains. In this regard, the model makes general predictions about the temporal coordination of multiple spike trains, which would be interesting to investigate experimentally.

One interesting property of this representation is that it results in a spike-like representation. In the resulting set of non-zero coefficients, not only is their value important for representing the signal, but also their relative temporal position, which indicate when an underlying event has occurred. This shares many properties with cochlear models. The model described here also has capacity to have an overcomplete representation at any given timepoint, e.g. a kernel basis with an arbitrarily large number of frequencies. These properties make this model potentially useful for binaural signal processing applications.

The effectiveness of this method for efficient coding remains to be proved. A trivial example of a shift-invariant basis is a delta-function model. For a model to encode information efficiently, the representation should be non-redundant. Each basis function should "grab" as much structure in the data as possible and achieve

the same level of coding efficiency for arbitrary shifts of the data. The matrix form of the model (12.4) suggests that it is possible to achieve this optimum by adapting the kernel functions themselves using methods for adapting overcomplete representations [16, 14]. Initial results suggest that this approach is promising. Beyond this, it is evident that modeling the higher-order structure in the coefficients themselves will be necessary both to achieve an efficient representation and to capture structure that is relevant to such tasks as speech recognition or auditory stream segmentation.

Acknowledgments

We thank Tony Bell, Bruno Olshausen, and David Donoho for helpful discussions.

References

[1] Atick, J. J. (1992). Could information-theory provide an ecological theory of sensory processing. *Network-Computation in Neural Systems*, 3(2):213–251.

[2] Attias, H. (1998). Blind source separation and deconvolution: the dynamic component analysis algorithm. *Neural Computation*, (10):1373–1424.

[3] Barlow, H. B. (1989). Unsupervised learning. *Neural Computation*, 1:295–311.

[4] Bell, A. J. and Sejnowski, T. J. (1995). An information maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7(6):1129–1159.

[5] Bialek, W., Rieke, F., de Ruyter van Steveninck, R. R., and Warland, D. (1991). Reading a neural code. *Science*, 252(5014):1854–1857.

[6] Cardoso, J.-F. (1997). Infomax and maximum likelihood for blind source separation. *IEEE Signal Processing Letters*, 4:109–111.

[7] Chen, S., Donoho, D. L., and Saunders, M. A. (1996). Atomic decomposition by basis pursuit. Technical report, Dept. Stat., Stanford Univ., Stanford, CA.

[8] Coifman, R. R. and Wickerhauser, M. V. (1992). Entropy-based algorithms for best basis selection. *IEEE Transactions on Information Theory*, 38(2):713–718.

[9] Comon, P. (1994). Independent component analysis, a new concept? *Signal Processing*, 36(3):287–314.

[10] Daubechies, I. (1990). The wavelet transform, time-frequency localization, and signal analysis. *IEEE Transactions on Information Theory*, 36(5):961–1004.

[11] de Boer, E. and Kuyper, P. (1968). Triggered correlation. *IEEE Trans. Biomed. Eng.*, 15(3):169–179.

[12] de Ruyter van Steveninck, R. R. and Bialek, W. (1988). Real-time performance of a movement-sensitive neuron in the blowfly visual system: coding and information transfer in short spike sequences. *Proc. R. Soc. B,*, 234:379–414.

[13] Gerstner, W. (1999). Spiking neurons. In Maass, W. and Bishop, C. M., editors, *Pulsed Neural Networks*, pages 3–53. MIT Press.

[14] Lewicki, M. S. and Olshausen, B. A. (1999). A probabilistic framework for the adaptation and comparison of image codes. *J. Opt. Soc. of Am. A: Optics, Image Science, and Vision*, 16(7):1587–1601.

[15] Lewicki, M. S. and Sejnowski, T. J. (1999). Coding time-varying signals using sparse, shift-invariant representations. In *Advances in Neural Information Processing Systems*, volume 11, pages 730–736. MIT Press.

[16] Lewicki, M. S. and Sejnowski, T. J. (2000). Learning overcomplete representations. *Neural Computation*, 12(2):337–365.

[17] Linsker, R. (1992). Local synaptic rules suffice to maximize mutual information in a linear network. *Neural Computation*, 4:691–702.

[18] Mallat, S. G. and Zhang, Z. F. (1993). Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415.

[19] Nadal, J.-P. and Parga, N. (1994). Nonlinear neurons in the low-noise limit: A a factorial code maximizes information transfer. *Network*, 5:565–581.

[20] Pearlmutter, B. A. and Parra, L. C. (1996). A context-senstive generalization of ICA. In *International Conference on Neural Information Processing*, pages 151–157.

[21] Simoncelli, E. P., Freeman, W. T., Adelson, E. H., and Heeger, D. J. (1992). Shiftable multiscale transforms. *IEEE Trans. Info. Theory*, 38:587–607.