Decoding Visual Input based on V1 Neuronal Activities with Particle Filtering [1]

Ryan Kelly and Tai Sing Lee

Center for the Neural Basis of Cognition & Computer Science Department
Carnegie Mellon University, Pittsburgh PA 15213, U.S.A.

Abstract

In this study, we investigated the use of particle filtering in reconstructing time-varying input visual signals based on Macaque V1 neurons' responses. A multitude of hypothesis particles are proposed for reconstructing the input stimulus up to time t. A prediction kernel (consisting of the first and second order forward Wiener kernels, derived by regression) is used to predict the neural response at time t based on the estimated input signals in the 200 ms prior to t. The fitness of this estimated response in predicting the measured response at time t is used to weigh the importance of the various hypotheses. The hypothesis particle space is collapsed by re-sampling over time. We find this method quite successful in reconstructing the input stimulus for 30 out of 33 V1 neurons measured. It out-performs the optimal linear decoder that we have experimented with in the past (Romero and Lee, 2002).

Key words: Particle Filtering, Neural Decoding.

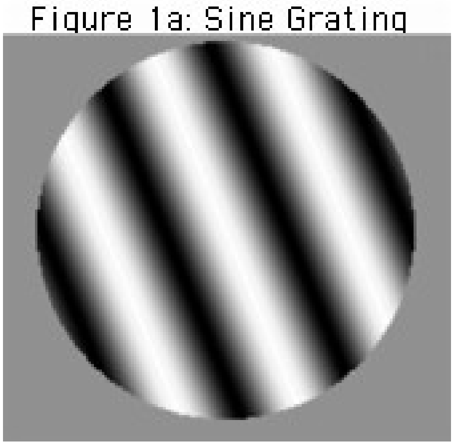
1 Introduction

Particle filtering has recently been used for estimating the arm trajectory of monkeys based on M1 neurons' responses [Gao et al. 2001] and for estimating the place field of hippocampus neurons [Eden et al. 2002]. Here, we applied a similar technique to recover the input stimulus signal based on the spike activities of V1 neurons in awake macaque monkeys. We will briefly describe the experiment and the input stimulus, and the results of the forward kernels, which are documented in an earlier CNS conference (Romero and Lee 2001, Romero et al. 2002). Then we will discuss the results of input signal reconstruction based on particle filtering.

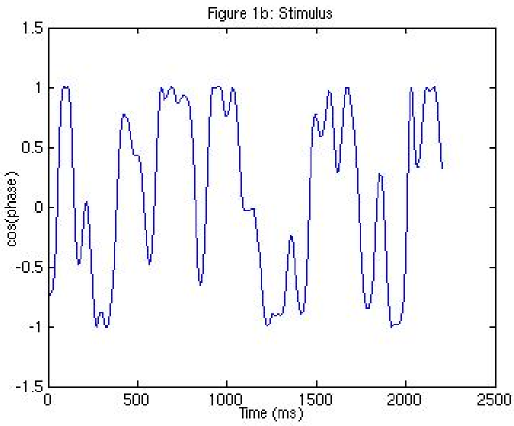
[1] This research was supported by This research was supported by NSF CAREER 9984706, NIH Vision Research core grant EY08098, and a NIH grant on the Pittsburgh Supercomputing Center.

2 The Experiment

We presented movies (2.2 second per trial) of a sinewave grating to the monkey while it is fixating on a red spot on the screen. The grating appeared inside a window of 5 degrees in diameter, the receptive fields of the tested cell were $< 1^o$ in diameter, typically 2^o to 4^o eccentricity away from the fovea. An example of the grating stimulus is shown in Figure 1a.



The orientation and spatial frequency of the sine wave are chosen to maximize the cell's response and modulation by the stimuli. We then restricted our stimuli to sine wave gratings that 'drifted' randomly in one dimension. The noise component is the phase $\phi(t)$ of the grating, which undergoes a random walk with step size in phase chosen from a Gaussian distribution, $\triangle\phi(t) \sim N(0, \sigma)$. We also introduce some temporal correlation to the stimuli by applying a low-pass filter to a true Gaussian random variable. This removes the high frequency components of the input signal and creates the desired temporal correlation. Since the phase has a discontinuity at 2π and 0, we applied a cosine transform to $\phi(t)$ so that the input signal $x(t) = \cos\phi(t)$ is a continuous signal. An example of the signal $x(t)$ is shown in Figure 1b.



10 random sequences and 2 repetitions of one sequence were presented in each block, for a total of 40 blocks, giving us 400 trials of unique random sequences and 80 trials of a particular stimulus sequence. The forward kernels

are recovered from the 400 training trials and used to predict the responses in the 80 repeated trials.

3 Forward kernels

The cell's transfer function with memory length L is given by by h such that the response y can be predicted by convolution of h and the input $x(t)$,

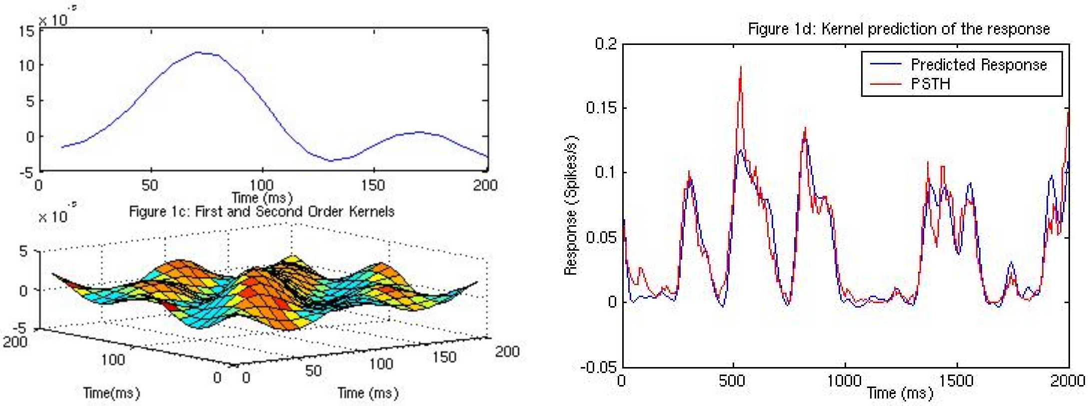
$$y(t) = h_o + \sum_{\tau=1}^{L} h_\tau x(t-\tau) + \sum_{\tau_2}^{L} \sum_{\tau_1}^{L} h_{\tau_1,\tau_2} x(t-\tau_1) x(t-\tau_2)$$

The forward kernels are derived by the regression technique. The standard solution is $H = (X'X)^{-1}X'Y$. Because of the correlations in the input signal $x(t)$, the matrix $(X'X)$ is ill conditioned. Instead of directly inverting this matrix, we use singular value decomposition:

$$USU' = X'X$$

where $US^{-1}U' = (X'X)^{-1}$ and S is a diagonal matrix. We include only the first n largest dimensions as ranked by their eigenvalue, where n is chosen to account for 99% of the variance in X.

Figure 1c shows an example of the first and second order forward Wiener kernels and Figure 1d shows how well they can be used to predict the response PSTH $y(t)$ of the repeated trials.



4 Particle Filtering

Particle filtering can then use the forward Wiener kernels to estimate the input signals as follows,

First, we compute the transitional prior on how the signals tend to move from the data in the training trials and construct a prior probability table $P(a, b)$ which is the probability of the stimulus value b at some moment, given that the previous stimulus value was a.

Given this prior and a post stimulus time histogram for the test (repeated) trials, we proceed with signal estimation using the following adaptation of the particle filtering method. We maintain a fixed number of particles, and each particle is associated with a vector of stimulus values. At each time step, we perform filtering and propagation on the particles.

During filtering, we obtain likelihoods (weights) for the particles. Each particle is passed through the kernels, and for each a response value $\hat{r}$ is predicted. We have the actual response value r at the time step, and the weights are computed as the distance between the predicted response value and the actual value.

$$w_i = \frac{e^{-(\hat{r_i}-r)^2/\sigma^2}}{\sum_j e^{-(\hat{r_j}-r)^2/\sigma^2}}$$

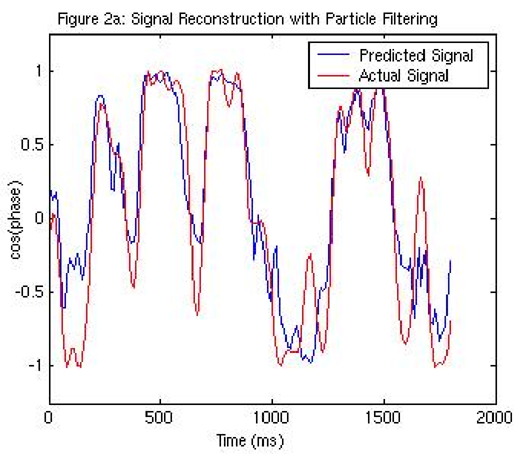
Particles which predict a response close to the actual response will be highly weighted. σ is a parameter which directly affects the relative weights of likely and unlikely particles. After the weights are computed, the particles are resampled with respect to these weights.

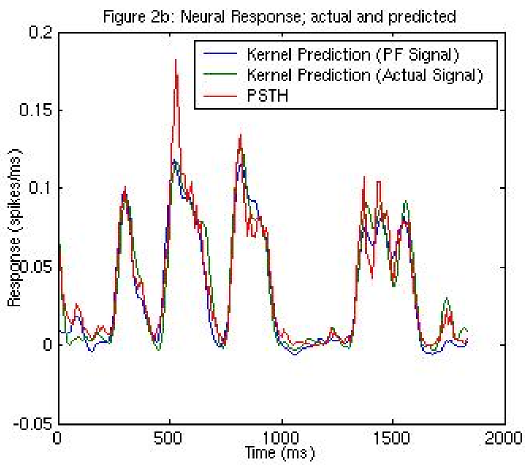
With the resampled particles, each particle has a value chosen for the stimulus at the next time step. These values are chosen independently and randomly with respect to the distribution P above. The process is repeated for all time steps, and the surviving particle is the recreated stimulus.

5 Results and Discussion

We find the particle filtering technique, even at its current preliminary implementation, is quite effective for this purpose. Figure 2a compares the reconstructed signal and the input signal demonstrating the effectiveness of the signal.

Figure 2b shows the estimated neural response based on the reconstructed signal passing through the forward kernels. The kernel's predictions are almost identical between the particle filtering signal and the actual signal.





Among data of the 33 neurons analyzed, we found that the method works very well for 30 of the neurons, the percentage errors for the 30 neurons are shown in Figure 3. The errors are a sum of squares computation of the residuals.

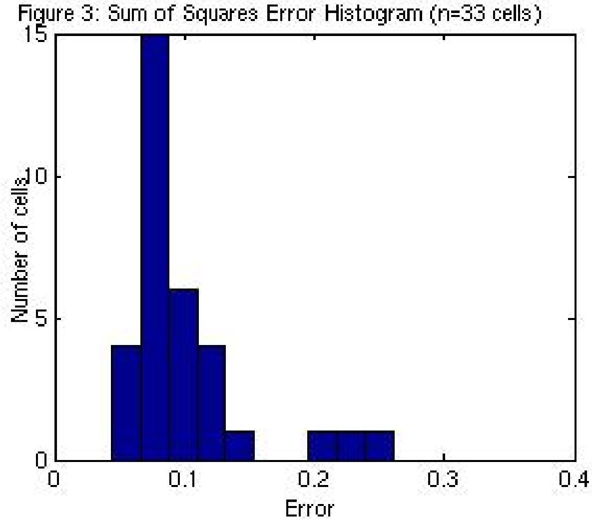
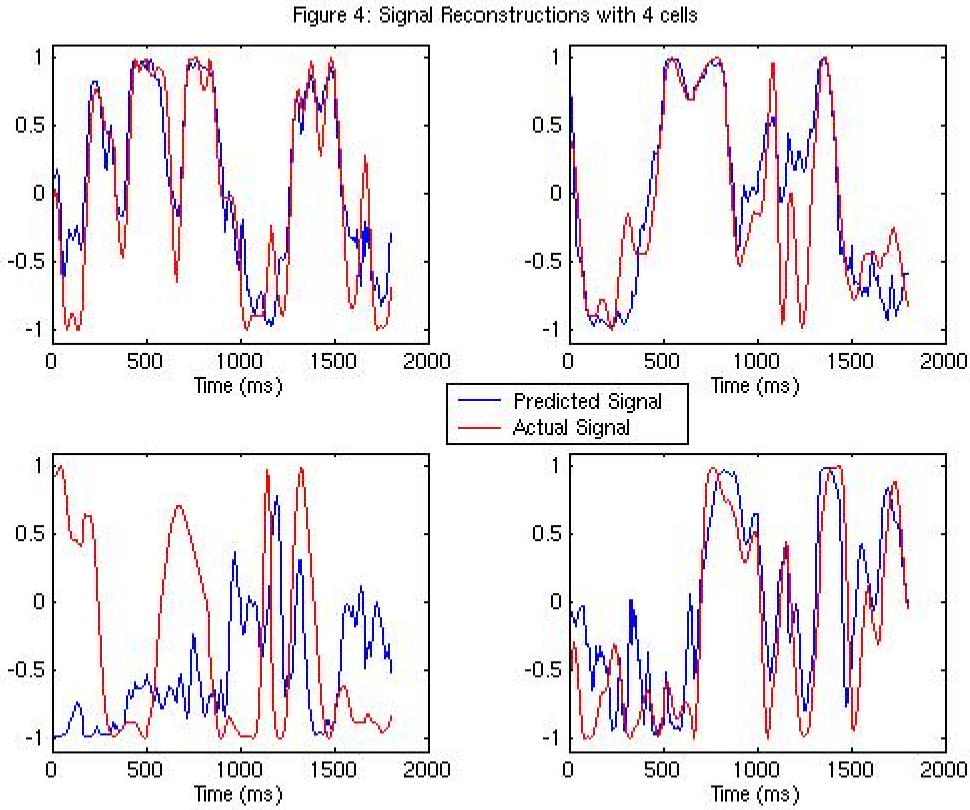


Figure 4 shows some reconstruction examples of the good cases (top) and the worst cases (bottom).



The particle filtering method is sensitive to several parameters. One is the number of particles used, and the other is the level of uncertainty σ in evaluating the weight.

We found the accuracy in the prediction improves as the inverse of the number of particles, with performance hitting asymptope around 1000 particles. This is the number of particles used in all displayed reconstructions.

σ affects the rate at which the particles converge on stimulus values. If σ is too large, all particles will become equally likely, while if σ is too small, only a few few particles will survive each time step. Ideally, the particles will converge on a value for a number of time steps equal to the kernel's length. The optimal value for σ could be found empirically by running sets of trials repeatedly to find the average error for different values of σ. This optimal value was used in all reconstructions.

References

[1] E.N. Brown, L.M. Frank, D. Tang, M.C. Quirk, M.A. Wilson, A statistical paradigm for neural spike train decoding applied to position prediction from ensemble firing patterns of rat hippocampal place cells. J. Neurosci, 18(18) (1998) 741-25.

[2] A. Doucet, N. de Freitas, N. Gordon, (Eds) *Sequential Monte Carlo Methods in Practice* (Springer-Verlag, New York, 2001).

[3] U.T. Eden, L.M. Frank, R. Barbieri, E.N. Brown, Particle filtering algorithms for neural decoding and adaptive estimation of receptive field plasticity. CNS 2002. Neural computing, in press.

[4] Y. Gao, M.J. Black, E. Bienenstock, S. Shoham, J. Donoghue, Probabilistic inference of arm motion from neural activity in motor cortex. Advance in neural information processing systems 14, The MIT Press, 2002.

[5] M.J. Korenberg, Identifying nonlinear difference equation and functional expansion representations: the fast orthogonal algorithm, Ann. Biomed. Eng. 16 (1988) 123-142.

[6] F. Rieke, D. Warland, R.D. Ruyter van Steveninck, & W. Bialek, *Spikes: Exploring the Neural Code*, Cambridge, MA: MIT Press, 1997.

[7] R. Romero, T.S. Lee, Spike train analysis for single trial data. Neurocomputing 44-46 (2002) 597-603.

[8] R. Romero, Y. Yu, P. Afhsar, T.S. Lee, Adaptation of temporal receptive field of macaque V1 neurons, neurcomputing, in press.

[9] G.B. Stanley, F.F. Li, and Y. Dan, Reconstruction of natural scenes from ensembles responses in the lateral geniculate nucleus, J. Neuroscience 19(18) (1999) 8036-8042.