STRFPAK: A Spatio-temporal Receptive Field Estimation Software

Theunissen Lab and Gallant Lab
University of California at Berkeley

March 19, 2003

Abstract

STRFPAK is a Matlab toolbox for estimating the linear/nonlinear stimulus-response transfer function of a sensory neuron. The resulting spatio-temporal receptive field (STRF) provides a quantitative description of neural filtering properties that can be used in subsequent computational modeling studies. The estimation techniques implemented by STRFPAK are quite general. Several algorithms are provided for estimating both linear and nonlinear STRFs from responses to either simple or complex stimuli, including natural signals. This documentation will describe the motivation and philosophy behind its design and provide details on how to use it.

Contents

List of Figures

Chapter 1

Introduction

This manual documents version 0.1 of STRFPAK, the Spatio-Temporal Receptive Field Estimation Software. STRFPAK is designed as a Matlab Toolbox with the graphical user interface for ease of use and understanding. STRFPAK estimates (non)linear stimulus-response transfer function of a sensory neuron from their responses to arbitrary stimuli such as auditory vocalizations or natural visual scenes [1]. The generalized reverse correlation techniques are implemented in STRFPAK to estimate STRF. The method used here should be useful for determining what aspects of natural signals are represented by sensory neurons.

For more information regarding computational method implemented in STRFPAK, see the paper "**Estimating spatio-temporal receptive fields of auditory and visual neurons from their responses to natural stimuli**" by F.E. Theunissen, S.V. David, N.C. Singh, A. Hsu, W. E. Vinje and J.L. Gallant, 2001 [1]. The electronic version of this paper can be obtained by email request to the authors.

Chapter 2

Design Philosophy

Graphical visualization is a standard technique for facilitating human comprehension of complex phenomena and large volumes of data. The study of neuron behavior is extremely complex, involving measuring and characterizing how stimulus attributes, such as light or sound intensity etc, are represented, and how neurons respond to a wide variety of stimuli. It is natural to use visualization techniques to gain insights into how input data look like, and how good we can fit and predict neuron responses from our given input data. The main window of STRFPAK shown in Figure 3.1 clearly pictures overall flow from getting input to do estimation, then to do fitness/prediction and do validation.

The three principal goals in designing STRFPAK are highly concerned: ease of understanding, ease of use and portability.

2.1 Ease of Understanding

STRFPAK is designed as a software tool with graphical user interface for ease of understanding and use. Since the purpose of graphic interface is to facilitate human understanding, it is imperative that the visual displays provided be as intuitively meaningful as possible. From the main window of STRFPAK, the basic funationality of STRFPAK is easily understood: getting input data, giving visual display of input data, estimating STRF from input data, displaying intermediate results of stimulus statistics and estimated STRF, predicting neuron response on different data set by using estimated STRF and validating goodness of fitting. The **help** button and self-comment field are provided for every window of STRFPAK. They convey the detailed information of each figure, chart or diagram shown on the window.

2.2 Ease of Use

One of the main purposes of software tools is to relieve tedium, not to promote it. Through the use of color, pop-up menu, editable text field, mouse and menu oriented user interface, STRFPAK is designed to keep the user's learning curve easy. It also provides interactive tracing window when the large execution takes place so that the user can easily gain insight into how and where the calculation procedure goes. The dialog windows are also implemented for showing detailed warning, error and confirmation messages.

2.3 Portability

STRFPAK is implemented using Matlab programming language. It can run on any operating systems that support Matlab and the X Window System. Although STRFPAK is effective in color, it also works on monochrome and grayscale monitors. For the future version, we propose to develop standalone executable program that will be independed of Matlab environment.

Chapter 3

Using STRFPAK

STRFPAK provides an important framework for estimating STRFs from a variety of stimuli and neuron responses. The first version of STRFPAK focuses on basic features of STRF estimation. For example, the user can visually display any stimulus and response data, see how good the estimated STRF works by doing prediction on different data sets and quantity estimated STRF by computing information values and correlation coefficients of prediction and real response. STRFPAK can also be easily downloaded and installed to any operating systems that support Matlab and X Window system. In this chapter, we first give instruction on how to install STRFPAK. Then we use two examples to explain what problems we can solve and what analysis we can do by using the STRFPAK. We also show how STRFPAK works by following the flow provided in the main window.

3.1 Download and Install STRFPAK

STRFPAK is supported by grant from NIMH and developed by Theunissen Lab and Gallant Lab at University of California, Berkeley. It is a free software now. The user can go to download page: $http://strfpak.berkeley.edu$ to get source codes.

After downloading the Unix/Linux version STRFPAK to the directory of the user's choice, use **gunzip STRFPAK.tar.gz** to uncompress it, and then **tar xvf STRFPAK.tar** on the command prompt to untar it to the directory **STRFPAK**. For installing the Window Version STRFPAK, use **WinZip** software to unzip it and then install it directly to the directory.

For any problems or questions about downloading and installing STRFPAK, please refer to our STRFPAK FAQ page in the STRFPAK web site.

3.2 Examples of Auditory and Visual Systems

The demo data for the examples are included under **STRFPAK**'s subdirectory **DemoData**. These data are from modeling experiment and real experiment.

3.2.1 Example of Auditory System

This auditory example data are from experiments done by the Theunissen Lab at University of California,Berkeley. Data were obtained from "con" stimuli for auditory neurons. The one-dimensional sound pressure waveform $s(t)$ is first transformed into a spectrographic representation, $s(x,t)$, which is the amplitude envelope of the sound in a set of frequency bands centered at x. In this example, the spectrographic representation uses 62 frequency bands of 125 Hz bandwidth, spanning the frequency range from 250 to 8000 Hz. The amplitude envelope in each band was sampled at a frequency of 1 kHz.

3.2.2 Example of Vision System

The vision example data are from experiments done by Gallant Lab at University of California, Berkeley. The stimuli class is a natural vision movie which simulates natural viewing of a static natural scene. It was constructed by extracting image patches along a simulated eye scan path. The stimulus ensemble consists of a natural vision movie 50 second long and is displayed with frequency of 72 Hz. The spatial plane has been downsampled by 16×16 pixels in order to facilitate display.

3.3 Application of the STRFPAK to the Examples

Theoretical neuroscience studies the link between stimulus and response from two opposite points of view: neural encoding and neural decoding [7]. Neural encoding refers to the map from stimulus to response and neural decoding refers to the map from response to stimulus. STRFPAK is a useful software tool for studying neural encoding. For example, the second-order statistics of natural stimuli are calculated in STRFPAK for studying properties of stimuli. STRFPAK can also be used to determine what aspects of stimulus are represented by sensory neurons, reveal the response properties of these neurons, and predict neuron responses to other stimuli. STRFPAK also implements information theory to estimate how the information is conveyed about stimuli by spike sequences and predicted responses.

Using the example data, we will see how STRFPAK works to solve above problems: studying properties of the stimuli, estimating the spatio-temporal transfer function of sensory neurons, predicting neuron response on different type of stimuli, and validating goodness of estimated STRF.

3.3.1 Main Window

As mentioned above, STRFPAK is implemented using the Matlab programming language. To start the STRFPAK, MATLAB needs to be started first. Then type the following command at the Matlab prompt:

strfpak

Then the current STRFPAK directory is added to the Matlab path and the main window shown in the Figure 3.1 appears. The functionality of STRFPAK can be accessed using the buttons at the center of the main window. It includes four stages. Each stage contained in the main window is described as follows:

- **Get Input:** In this stage, user specifies the input data by clicking **Get Files** button. Then he/she needs to set the calculation parameters using the **Parameters** button. The input data can be displayed graphically by clicking the **Display Input** button.

- **Estimate:** In this stage, the second order statistics of the input stimulus get calculated and the STRF get estimated by clicking **Calculate** button. The stimulus auto-correlation matrix and the stimulus-response cross correlation can be displayed using the **Display StimStat** button. The estimated STRF and its related analysis can be displayed by clicking the **Display STRFs** button.

- **Predict:** In this stage, the estimated STRFs are used to predict neuron responses with the data files provided by the user for prediction. By clicking **Get PredFiles** button, the user selects the data files used for predication. Using the **Predict** button, the user gets the prediction from the estimated STRFs and stimuli from the user input data files. The prediction results can be displayed graphically using the **Display PredPSTH** button.

- **Validate:** In this stage, the information values and correlation coefficients are calculated by clicking **Validate** button to check the goodness of fit. The results are displayed using the **Display Info** button. The best estimated STRF based on predicted information values is displayed using the **Display BestStrf** button.

The status window at the bottom of the main window describes the current stage of execution. The **Help** window pops up by pressing the **Help** button. The window is closed by pressing the **Close** button.

3.3.2 Get Datafiles Window

For portability, STRFPAK requires the input data format either in ASCII or Matlab binary format, which typically has *.txt*, *.dat* or *.mat* extension. Thus, the raw data from experiment needs to be preprocessed to have these data formats. A stimulus data file contains a spatio-temporal description of a stimulus, $s(x,t)$. A response data file contains multiple trials of spike trains, $r(trials,t)$. For example, in the auditory system, $s(x,t)$ represents the time-varying amplitude of sound in a frequency band centered at x. In the visual system, $s(x,t)$ represents the light intensity as a function of position and time. For better prediction, the larger data sets are prefered. Figures 3.2 and 3.3 show the **Get Datafiles** Windows for the auditory example and the visual example respectively.

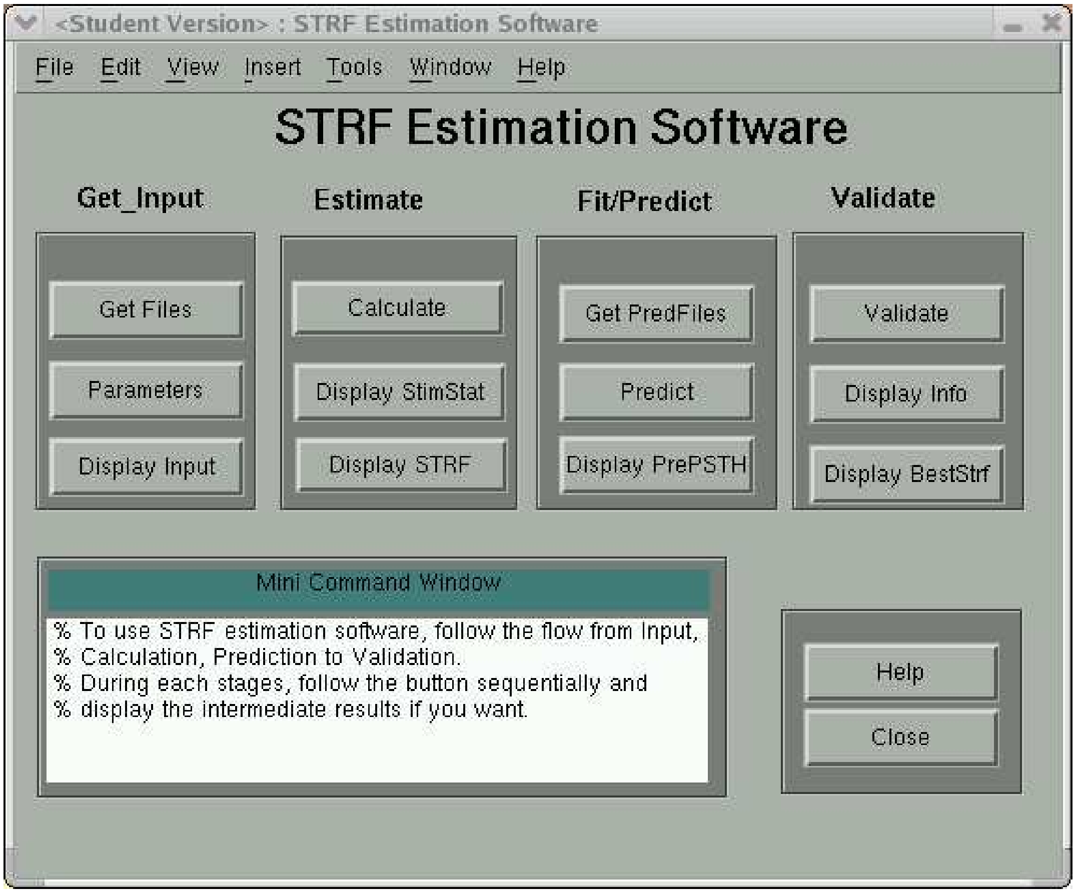

Figure 3.1: The Main Screen of STRFPAK

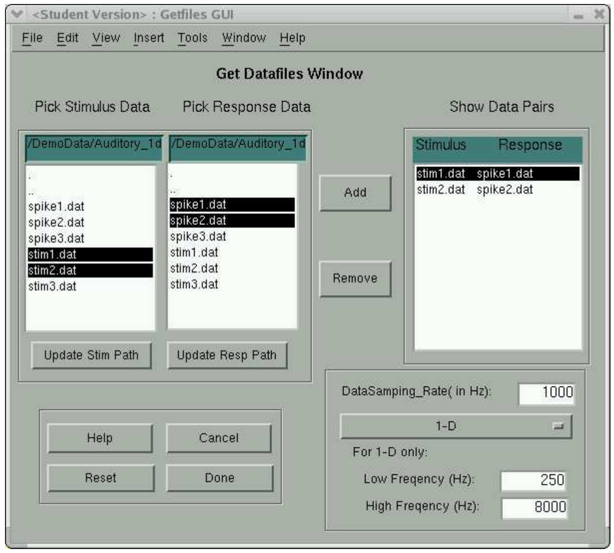

Figure 3.2: Get datafiles window for the auditory example

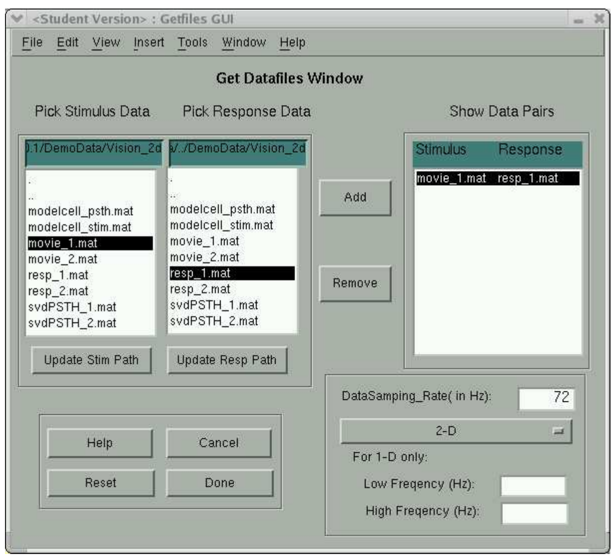

Figure 3.3: Get datafiles window for the visual example

In this window, the user can choose stimulus and response input from the top left panel of the window. In this panel, there are two list boxes for stimulus files and response files. The list boxes display the files under current directories of the stimulus and the response. The user can interactively update these directories for stimulus and response list boxes by clicking **Update Stim Path** or **Update Resp Path** buttons. When user clicks on a list item in the stimulus or the response list box, one of the following happens:

- If the item is a file, the file is selected.
- If the item is a directory, the GUI reads the contents of that directory into the list box.
- If the item is a single dot (.), the GUI updates the display of the current directory.
- If the item is a double dot (..), the GUI changes to the directory up one level and populates the list box with the contents of that directory.

To select multiple data sets, hold down the CTRL or Shift key while clicking selections. After selecting stimulus files from the stimulus list box and response files from the response list box, the user can click **Add** button to check if the file type is allowed and save them as global variables for later analysis. They will show on the **show data pairs** list box if data format is OK. The user can remove the selected data files by pressing **Remove** button.

After the selection has been done, the user needs to specify data parameters in the right bottom panel of the window by clicking the pop-up menu **Please Choose Spatial Domain** and by filling in the editable text field. These parameters include data sampling rate (in Hz), dimensionality of spatial domain, and low and high frequency (in Hz) for $1D$ spatial domain. For the auditory example, data sampling rate is 1000 Hz, the spatial domain is a frequency band so the dimensionality of the spatial domain is set as $1D$. The low frequency is 250 Hz and the high frequency is 8000 Hz since the frequency band expands from 250 to 8000 Hz in the above description of the auditory experiment. For the visual example, the data sampling rate is 72 Hz. The dimensionality of spatial domain is $2D$ since stimulus is represented as position at time t.

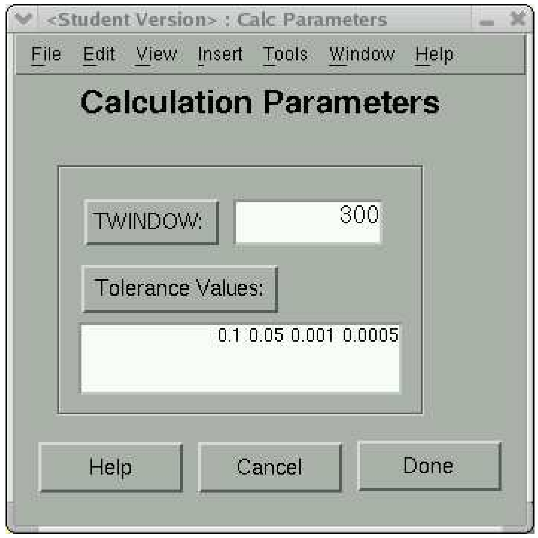

Figure 3.4: Calculation parameter window for the auditory example

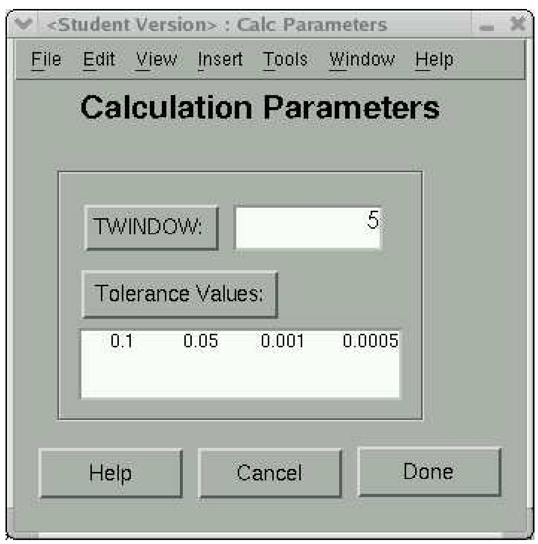

Figure 3.5: Calculation parameter window for the visual example

3.3.3 Calculation Parameter Window

Figures 3.4 and 3.5 show the **Calculation Parameter Window** for the auditory example and the visual example. There are two calculation parameters required for STRFPAK: **TimeLag** and **Tolerance value**. By clicking the **TimeLag** or **Tolerance value** button, a small information window will show up to explain the meaning and the format for the input text field. The **TimeLag** is a time-lag used for computing the stimulus auto-correlation and the stimulus-response cross correlation. For example, since the bin size for the above auditory data is 1 ms, the time course covers a period of ± 300 ms if we set **TimeLag** as 300. For the visual example, the bin size is 14 ms, the time course covers a period of ± 70 ms if we set **TimeLag** as 5. If the larger range is needed, **TimeLag** is needed to be larger value.

Tolerance value is a list of real values used to select eigenvalue cutoff for pseudo-inverse of the stimulus auto-correlation matrix. Its value should be less than 1. For example, if **Tolerance value** is 0.1, that means top 90% eigenvalues are used for calculating inverse of the auto-correlation matrix. The **Tolerance value** is usually set as a list of values to check the sensitivity of the final results to the tolerance values. In the above examples, we have set $Tol\ val$ as 0.1 0.05 0.001 0.0005.

3.3.4 Display Input Window

For convenience, STRFPAK provides graphical display of the input data in the **Display Input** window. The user can check whether the selected input data is good or not for later estimation and analysis. Figure 3.6 shows spectrographic representation of the stimulus, 10 trials of neuron response and its post-stimulus time histogram (PSTH) for the auditory example.

Figure 3.7 shows first 12 frames of natural scenes used in the above visual example, one single trial (since we have only one trial here) and its PSTH.

In the figures, the left panel of the window is graphical display of the stimulus, 10 trials of spike trains and the smoothed PSTH.

- Plot of stimulus file:
  If the dimensionality of the spatial domain is $1D$, x-axis is time (in seconds) and y-axis is frequency in Hz. If the dimensionality of spatial domain is $2D$, the first 12 video frames show up. For the current version of STRFPAK, we only show the first 12 frames.

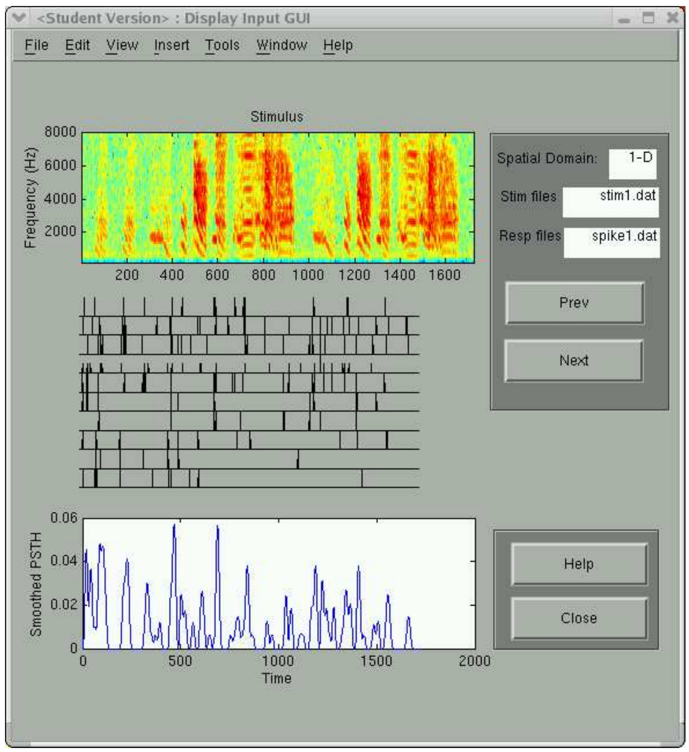

Figure 3.6: Display auditory input data

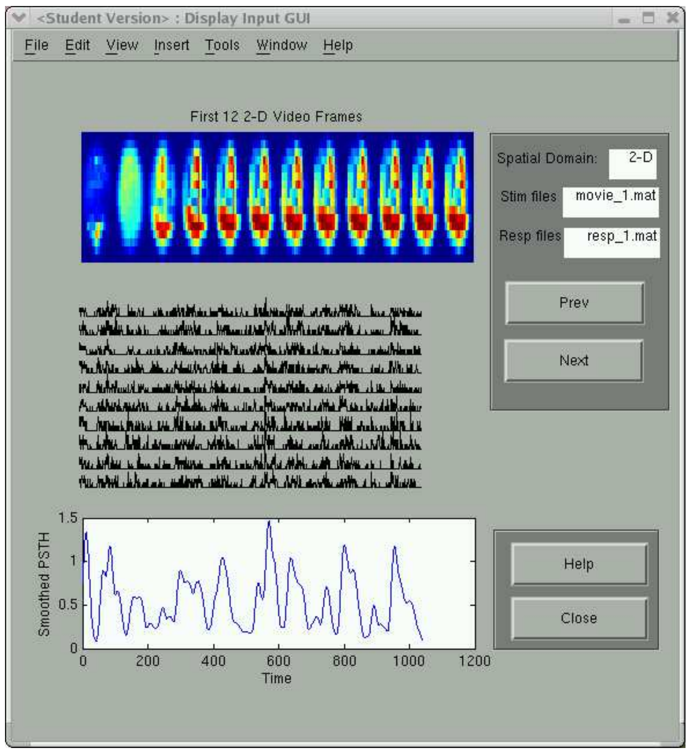

Figure 3.7: Display visual input data

- Plot of spike trains (in the sequence of time):
  Note: We only display 10 trials for the current version of STRFPAK. If we have more than 10 trials, we display the first 10 trials.

- Plot of smoothed psth (in time domain):
  Note: If we have single trail spike train, we take its psth as itself. Otherwise, we compute the average of multiple trials as their psth. The smoothed psth is computed as a covariance between raw psth and a Hanning window.

The right panel shows information of the figures displayed in the left panel. The user can choose **prev** or **next** button to display different data sets if multiple data pairs are selected. The stimulus text field and the response text field and the spatial-domain text field show the corresponding input data pair files are displayed in the left panel plots.

3.3.5 Estimate

STRFPAK implements the generalized reverse correlation method described in Theunissen et al 2001 **??** to estimate the STRFs of sensory neurons from their responses to complex stimulus ensembles. Since a complete description of the second-order statistics of the stimulus ensemble is required for estimation, STRFPAK first computes the stimulus auto-correlation matrix and the stimulus-response cross correlation vector. The auto-correlation matrix of the stimulus C_{ss} and the stimulus-response cross-correlation vector C_{sr} are described as follows **??**.

$$C_{ss} = \begin{pmatrix} c_{0,0} & \cdots & c_{0,M-1} \\ \vdots & \ddots & \vdots \\ c_{M-1,0} & \cdots & c_{M-1,M-1} \end{pmatrix}$$

and

$$C_{sr} = < sr > = \begin{pmatrix} < s[t-0]r[t] > \\ \vdots \\ < s[t-NM+1]r[t] > \end{pmatrix}$$

where $c_{i,j}$ denotes the correlations between spatial dimensions i and j for all the relevant time delays. N is the length of time dimension and M is the number of spatial parameters.

The estimated STRF from the linear mean-square estimation $< (\hat{r} - r)^2 >$ is given as follows:

$$h = C_{ss}^{-1} C_{sr}$$

where h is the estimated STRF. STRFPAK then normalizes the cross-covariance matrix between the stimulus and the response by auto-covariance matrix of the stimulus to get the estimated STRF. For error estimation analysis, the Jackknifed STRFs are also calculated if multiple data sets are selected. For the Jackknifed error estimation techniques, please refer to [4].

When the calculation are done, the small "Done Estimation" window appears.

3.3.6 Display Stimulus Statistics Window

As mentioned above, the second-order statistics of the stimulus ensemble has been computed in the Calculation stage. This window visually display the above results. Three display options are provided: **displaying stimulus auto-correlation matrix**; **displaying raw and smoothed stimulus-response cross-correlation**; and **displaying** $2D$ **raw and smoothed cross-correlation in a separate window**. These options are implemented as a popup menu in the top right of the window.

Figure 3.8 shows the stimulus auto-correlation matrix in the space-time domain for the auditory example. Each entry in the matrix corresponds to the temporal cross-correlation of the sound amplitude in two different frequency bands. It is organized with lowest center frequency of the band at the top left corner and the highest frequency at the bottom right (see **??** for details). Figure 3.9 shows raw stimulus-response cross-correlation (also called Spike-triggered Average (STA)) and smoothed stimulus-response cross-correlation for the visual example. From this figure, the significant Gabor structure in the STA occurs in the sixth frame.

3.3.7 Display STRFs Window

The **Display STRFS window** provides a graphic display of the estimated STRF, STRF with gain on two axes and temporal modulation, and estimated STRF with Spike-Triggered Average (STA). These options can be selected by clicking the popup menu in the top right of the window.

Figure 3.10 shows estimated STRF with its STA for the auditory data. The upper figure shows the estimated STRF and the lower figure shows the

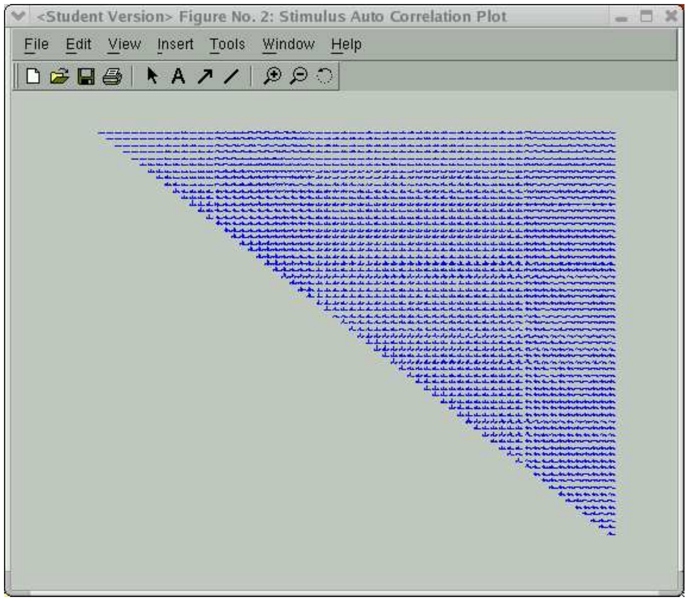

Figure 3.8: Display stimulus auto-correlation for the auditory example

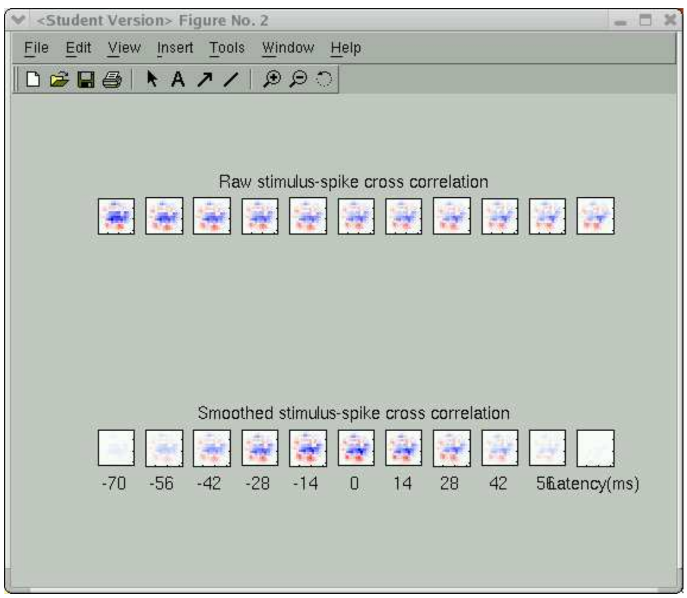

Figure 3.9: Display stimulus-response cross-correlation for the visual example

smoothed STA. The **Next** and **Prev** buttons help to show the estimated STRF for the different **Tol Value**. The results of the STRF estimation for the visual neuron example and the STA are shown in Figure 3.11. From the figure, we see that the STA closely matches estimated STRF for tolerance value 0.0005.

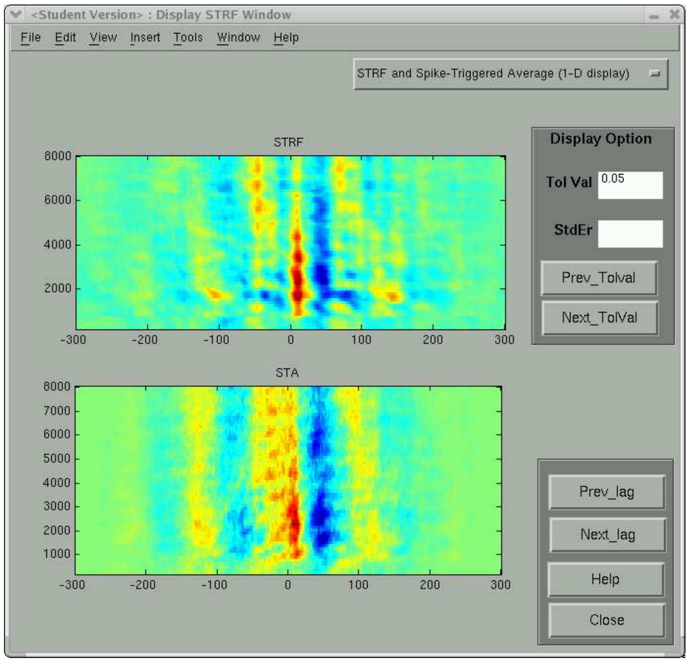

Figure 3.10: Display estimated STRF and STA for the auditory example

3.3.8 Get Datafiles for Prediction and Validation

There are two options to compute prediction and validation in the current version of STRFPAK. These options are implemented by setting the value of parameter *watch flag*. After clicking the **Get PredFiles** button in the main window, a window appears waiting for the input value of the *watch flag* from the user. If *watch flag* is set to 1, no new data pairs are required since the estimated Jackknifed STRF is used on the unused data pairs. If *watch flag* is set to 0, the **Get Datafiles for Prediction** window is going to show up. New stimulus-response data pairs have to be selected by following the similar procedures to the previous **Get Datafiles** window. For how to select new

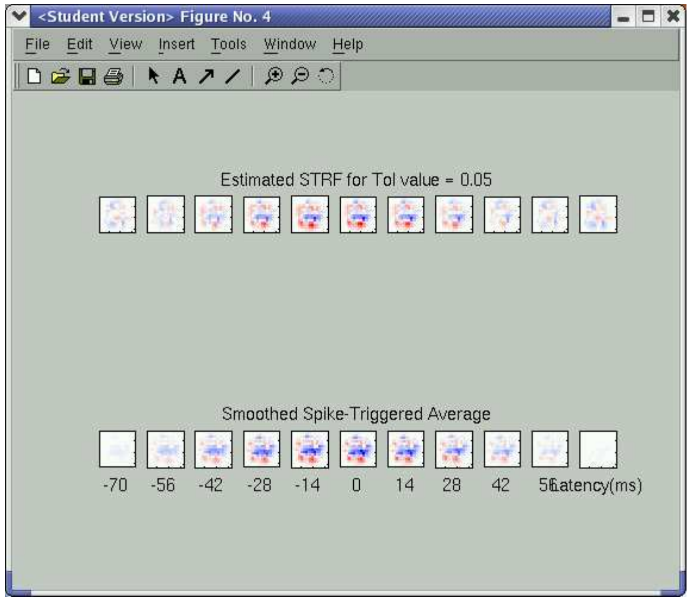

Figure 3.11: Display estimated STRF and STA for the visual example

data for prediction and validation in **Get Datafiles for Prediction and Validation** window, please refer to the instruction for the **Get Datafiles** window.

3.3.9 Predict

STRFPAK predicts neuron response by using stimulus which is provided by the user and the estimated STRF from above calculation. Whether estimated Jackknifed STRF or estimated STRF is used are based on the value of $watch\ flag$. The equation is given as:

$$\hat{r}[t] = \sum_{i=0}^{MN-1} h[i]s_t[i]$$

where the $h[i] = [h_0, ..., h_{NM-1}]^T$ vector is the linear coefficient describing the linear estimator and $s_t[i]$ are stimulus for prediction. Both the stimulus and the response are sampled at particular points in time t. In this stage, STRFPAK also reorganizes multiple trials of spike trains into two groups and then compute PSTH for each group. The results of these two PSTHs and predicted PSTH will be used for later validation.

When all calculations are done, small "Done Prediction" window appears.

3.3.10 Display Predicted PSTHs Window

After the prediction stage, we have a predicted PSTH and two PSTHs from the actual neuron response. **Display Predicted PSTHs Window** provides a graphical display of these results.

Figure 3.12 shows predicted results for the auditory example at $TolValue = 0.05$. The predicted results for the visual example are shown in Figure 3.13. The top left panel in the window shows the graph of the input stimulus used for prediction. If the spatial domain of the input stimulus is $1D$, $x\ axis$ is time and $y\ axis$ is frequency in Hz. If the spatial domain of the input stimulus is $2D$, it only shows first 12 $2D$ video frames in the current version of STRFPAK. The bottom left panel shows the predicted PSTH together with the two PSTHs from the input response. The user can easily see the goodness of fit from this plot. The right panel of the window is for information display. The tolerance value used for above plots is given in the text field of the **Tol val**. If more than one Tol values are used for calculation, different Tol. values can be shown by pressing **Next Tol Val** and **Prev Tol Val** buttons. The names of stimulus file and response file used for the plots in the left panel are shown in **pred stim file** field and **pred resp file** field. If more than one stimulus-response files are used for the prediction, different files can be shown by pressing **Next file** and **Prev file** buttons. **Help** and **Close** buttons are provided in the bottom right position of the window.

3.3.11 Validate

The **Display Predicted PSTH** window gives us visual comparison of the predicted response with the actual response. To quantify the goodness of fit of the estimated STRF, STRFPAK implements two measures: coherence and correlation coefficient. The coherence is a function of frequency and is given by:

$$\gamma^2(\omega) = \frac{< R(\omega)\hat{R}(\omega)^* >< R(\omega)^*\hat{R}(\omega) >}{< R(\omega)R(\omega)^* >< \hat{R}(\omega)\hat{R}(\omega)^* >}$$

Here $R(\omega)$ and $\hat{R}(\omega)$ are actual and predicted neuron responses at each temporal frequency, ω. An overall goodness-of-fit estimate, I, is obtained by integrating the coherence function. The lower bound of I is obtained if the

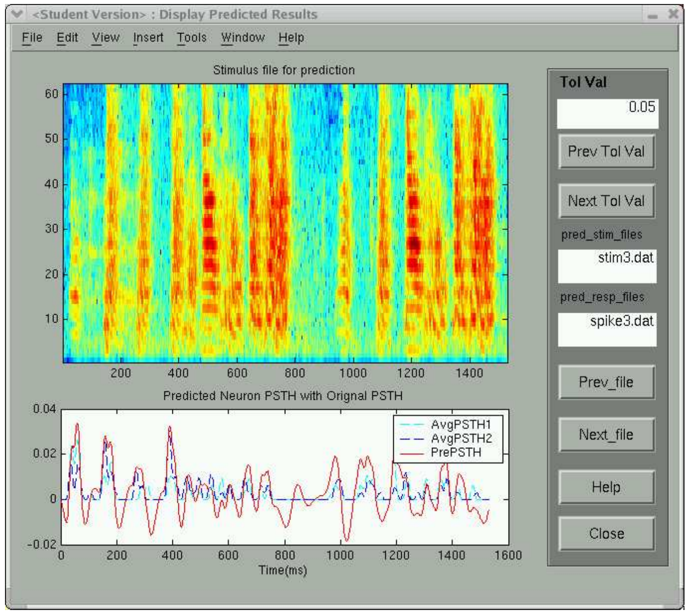

Figure 3.12: Display prediction results for the auditory example

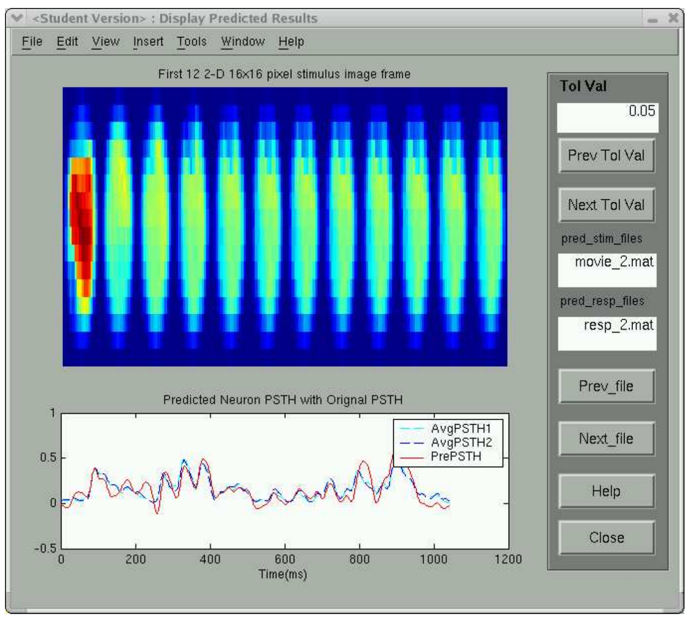

Figure 3.13: Display prediction results for the visual example

noise follows a Gaussian distribution and the upper bound is obtained if the neuron response is a Gaussian. For detailed derivation, please refer to Hsu's information theory paper(to be submitted).

Correlation coefficient(cc) between $r(t)$ and $\hat{r}(t)$ is calculated based on:

$$cc = \frac{< (r(t) - r\bar{(t)})(\hat{r}(t) - \hat{r}\bar{(t)}) >}{\sqrt{< (r(t) - r\bar{(t)})^2 >< (\hat{r}(t) - \hat{r}\bar{(t)})^2 >}}$$

Here $r(t)$ and $\hat{r}(t)$ are actual and predicted neuron responses. Since cc depends on the time bin that is used to obtain $r(t)$ from the PSTH, STRFPAK only compares cc between similar time windows **??**.

When all calculation are done, small "Done Prediction" window appears.

3.3.12 Display Information Value and Correlation Coefficient Window

Display Information Value and Correlation Coefficient window provides three display options: cc **display**, $info$ **display**, and **all the tol val vs. smoothing window**. These options are implemented as a popup menu in this window. For example, if the popup menu is chosen to show *cc display*, the predicated cc between the calculated $\hat{r}(t)$ and the actual $r(t)$, and the original cc between two PSTHs are shown in the Figure 3.14. Figure 3.15 shows the predicted coherence and the original coherence for the auditory demo data if the popup menu is chosen to show $Info$ display.

For the cc display option, x $axis$ in Figure 3.14 denotes smoothing filter width in the unit of ms. The x $axis$ in Figure 3.15 of the $Info$ display window is frequency in the unit of Hz. The green dot line in the $Info$ display window is the upper bound of information value and the red dot line is the lower bound. The tolerance value in the text field of **Tol val** is used for the predicted results displayed in the left panel. If more than one Tol value is used before, the user can use the **Prev tolval** or the **Next tolval** button to show the corresponding plots for the predicted results.

3.3.13 Display Best STRF Window

When the user clicks the **Display Best STRF** button in the main window, it sorts the estimated STRFs based on the predicted information values and

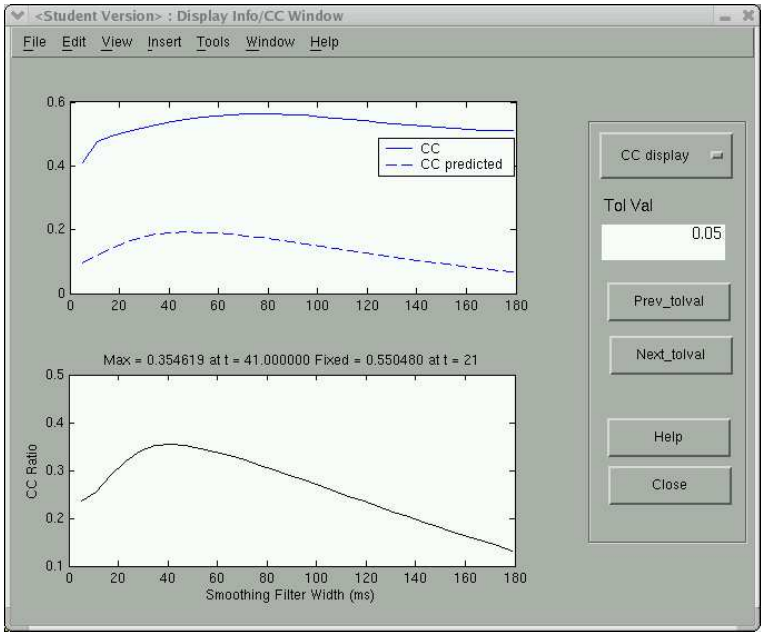

Figure 3.14: Display predicted cc and cc for the auditory example

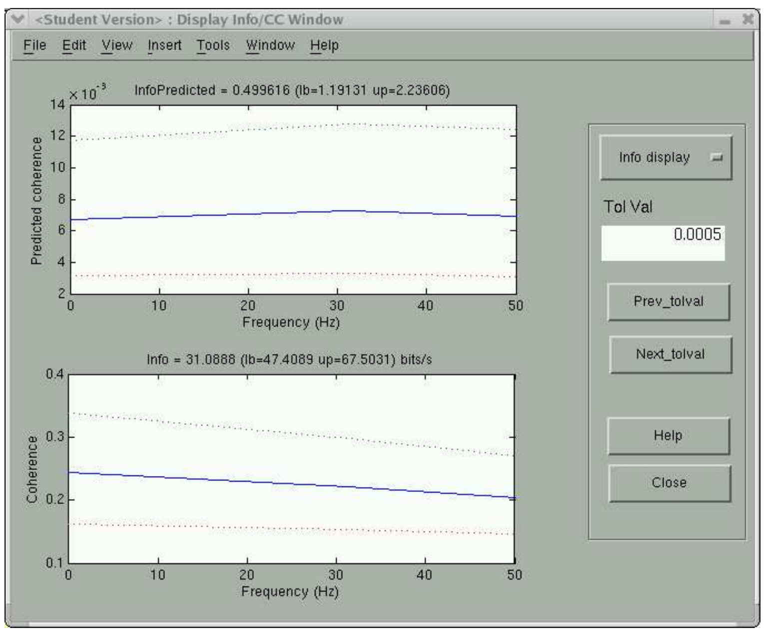

Figure 3.15: Display predicted coherence and coherence for the auditory example

chooses the best estimated STRF which has the largest predicted information value. It then displays the best estimated STRF in the left panel of the window and gives all the related information in the right text fields of the window. Figure 3.16 shows the best estimated STRF for the auditory example. The right panel in the window gives the tolerance value used for this STRF, the information value from the actual data, the predicted information value of the best estimated STRF, and the max $ccrPredicted/cc$ ratio. If the spatial domain of the stimulus file is $2D$, the best filter is shown as a list of $2D$ video frames.

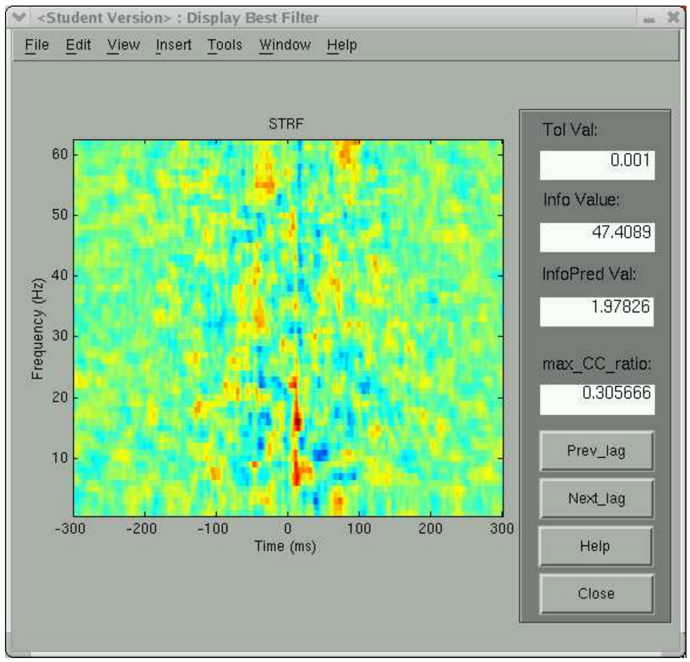

Figure 3.16: Display best estimated STRF for the auditory example

Chapter 4

Summary and Future Work

We have developed the first version of the spatio-temporal receptive field estimation software package, STRFPAK. This software is now available online at http://strfpak.berkeley.edu as a beta testing version. It estimates stimulus-response transfer function of a sensory neuron. The resulting spatio-temporal receptive field provides a quantitative description of the transformation between a time varying spatial stimulus and the neural response, which can be used in subsequent computational modeling studies. In the current version, we have implemented generalized reverse correlation technique and the Jackknifed error estimation algorithm to calculate the linear spatio-temporal receptive field. Two different measures, coherence and correlation coefficients, that quantify the estimated STRF's goodness of fit are also included in this version. We have also developed a graphic user interface with tutorial examples and help documents. The STRFPAK is implemented using Matlab programming language and organized as a Matlab tool box. It has been tested on Unix, Linux and Windows.

For future work, there are a number of areas proposed here. We mentioned above the limitation of data format that STRFPAK can handle is only ASCII or Matlab binary. For the next version, we plan to add the data preprocessing package for handling more general input format. The nonlinear estimation technique such as neuron network method is being developed and will be added to the future version of STRFPAK.

Bibliography

[1] F. E. Theunissen, S. V. David, N. C. Singh, A. Hsu, W. E. Vinje and J. L. Gallant, *Estimating spatio-temporal receptive fields of auditory and visual neurons from their responses to natural stimuli*, Network: Computation in Neural Systems 12 (2001) 289-316.

[2] W.E. Vinje and J.L. Gallant, *Sparse coding and decorrelation in primary visual cortex during natural vision*, Science 287 (2000) 1273-6.

[3] F.E. Theunissen, K. Sen and A.J. Doupe, *Spectral-temporal receptive fields of nonlinear auditory neurons obtained using natural sounds*, J. Neurosci., 20 (2000) 2315-31.

[4] D.J. Thomson and A.D. Chave, *Jackknifed error estimates for spectra, coherence and transfer functions*, Advances in Spectrum Analysis and Array Processing vol 1, ed S Haykin (Upper Saddle River, NJ:Prentice-Hall).

[5] Anne Hsu's coherence examination paper(to be submitted).

[6] S.V. David, W.E. Vinje and J.L. Gallant, *Natural image reverse correlation in awake behaving primates*, Soc. Neurosci. Abstr. 25, (1999) 1935.

[7] P. Dayan and L.F. Abbott, *Theoretical Neurosciense: Computational and Mathematical Modeling of Neural Systems*, The MIT Press, Cambridge, Massachusetts, 2001.

[8] P Marmeralis and V. Marmeralis, *Analysis of Physiological Systems: The White Noise Approach*, New York: Plenum, 1978.