VOLTERRA SERIES FOR ANALYZING MLP BASED PHONEME POSTERIOR ESTIMATOR

Joel Pinto [1,2], *G.S.V.S. Sivaram* [1,2], *H. Hermansky* [1,2], *M. Magimai.-Doss* [1]

[1] Idiap Research Institute, Martigny, Switzerland
[2] École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
{jpinto,sgarimel,hynek,mathew}@idiap.ch

ABSTRACT

We present a framework to apply Volterra series to analyze multilayered perceptrons trained to estimate the posterior probabilities of phonemes in automatic speech recognition. The identified Volterra kernels reveal the spectro-temporal patterns that are learned by the trained system for each phoneme. To demonstrate the applicability of Volterra series, we analyze a multilayered perceptron trained using Mel filter bank energy features and analyze its first order Volterra kernels.

Index Terms— Volterra series, multilayered perceptrons, speech recognition

1. INTRODUCTION

Multilayered perceptron (MLP) based acoustic modeling is being extensively used in the state-of-the-art automatic speech recognition (ASR) [1][2]. The MLP is trained as a phoneme classifier, and estimates the posterior probabilities of the phonemes conditioned on the input features. The estimates of posterior probabilities are used in ASR typically as local acoustic scores in hybrid hidden Markov model (HMM) - artificial neural network system [3] or as features (after logarithm and principal component analysis transformation) to a standard HMM - Gaussian mixture model system [4].

MLP based acoustic modeling has been shown to improve recognition accuracies in ASR. However, once trained, the MLP is typically not further analyzed. The estimated posterior probabilities are typically evaluated using (i) frame-level phoneme classification accuracy (ii) phonetic confusion matrix (iii) mutual information between the estimated posterior probabilities and its ground truth phonetic labels or (iv) the final speech recognition accuracy. While the above metrics indicate the goodness of the phoneme posterior estimates, none of them reveal any information on the spectro-temporal patterns that the trained system has learned.

One way to analyze the trained system is to treat it as a nonlinear black-box and present white Gaussian noise as input. The characteristics of the unknown system can be measured by cross-correlating the input white noise and the output of the system [5]. However, the three layered MLP based phoneme posterior estimator, which is typically used in ASR is simple enough to be analyzed analytically.

We formulate a framework to apply Volterra series [6] to analyze the trained MLPs. It is important to incorporate the feature extraction into this analysis because the identified Volterra kernels can then be interpreted as spectro-temporal patterns. The combined system is nonlinear and time-invariant, where the finite impulse response (FIR) filters used in feature extraction introduce memory and the activation functions in the MLP introduce nonlinearity. Volterra series has been used to model recurrent neural networks to analyze nonlinear properties of electronic devices [7]. The contributions of our work include (i) formulation of a framework to apply Volterra series to analyze MLPs estimating posterior probabilities of phonemes (ii) analytical identification of the Volterra kernels, (iii) addressing the effect of feature mean and variance normalization, and (iv) as an example, application of Volterra series to analyze an MLP trained on Mel filter bank energies

This work was supported by the Swiss national science foundation under the Indo-Swiss joint research program on keyword spotting (KEYSPOT) as well as the Swiss National Center for Competence in Research (NCCR) under the Interactive Multimodal Information Management (IM2) project.

2. PHONEME POSTERIOR ESTIMATOR

Fig. 1(a) is the block schematic of a typical phoneme posterior probability estimator, showing the feature extraction as well as the MLP classifier. Auditory analysis is a common stage across almost all feature extraction techniques. Short time Fourier analysis is performed on speech signal with an analysis window of typically 25 ms and a frame shift of 10 ms. Auditory filters that are equally spaced in Mel or Bark frequency scale are applied on the Fourier magnitude spectrum, and log energies in the auditory channels are computed.

The trajectories of the log energies from the auditory analysis are then processed by a linear time-invariant (LTI) system whose impulse response is decided by the feature extraction being used. For Mel frequency cepstral coefficients (MFCC), this system consists of discrete cosine transform (DCT), the FIR filters required to compute the delta and delta-delta coefficients, and the filters creating a temporal context of features. In the case of multi-resolution relative spectra (MRASTA) features [8], the LTI system consists of a bank of zero mean filters whose shape is that of either the first or second derivative of a Gaussian function. For Mel filter bank energy (MFB) features, the system consists of bank of time shifted Kronecker delta functions required to create a temporal context of features. From a mathematical perspective, the difference between the above feature extraction techniques is in the impulse response of the LTI system.

A three layered MLP with sigmoid nonlinearity at the hidden layer and softmax nonlinearity at the output layer is typically used. The MLP weights are trained to minimize the cross entropy between the estimated posterior probabilities and the phonetic labels.

3. VOLTERRA SERIES

An LTI system can be completely characterized by its impulse response function. Volterra series is an infinite series which can be used to express the input-output relationship in a nonlinear time-invariant system. Each term in the series is a multi-dimensional convolution between the input to the system and its Volterra kernels.

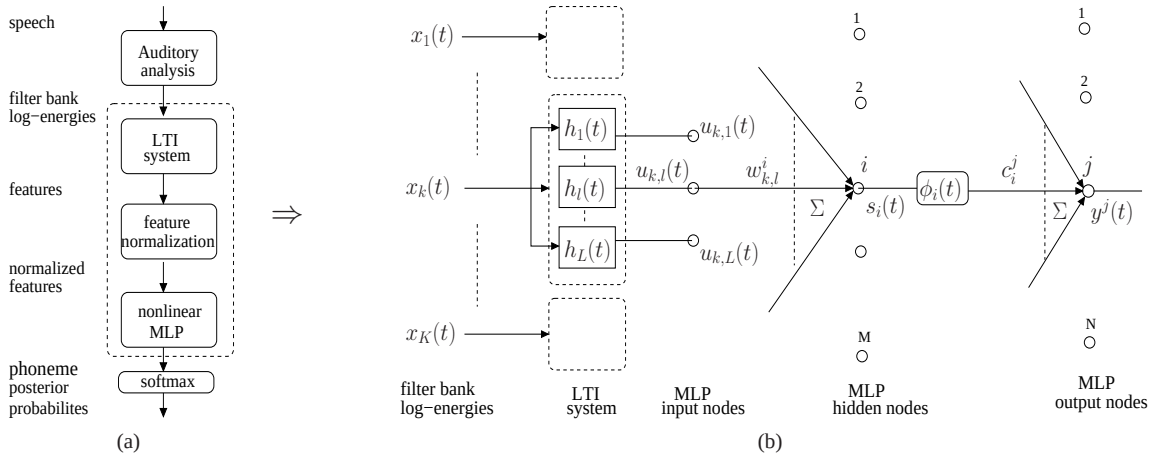

Fig. 1. *(a) Estimation of posterior probabilities of phonemes using an MLP. (b) Part of the system that is analyzed using Volterra series.*

The identified Volterra kernels completely characterize the nonlinear system. If $x(t)$ is the input to a nonlinear system and $y(t)$ its output, Volterra series expansion for the system can be expressed as

$$y(t) = \sum_{n=0}^{\infty} G_n\left[g_n, x(t)\right]$$

where, $\{G_n\}$ is the set of Volterra functionals, and $\{g_n\}$ is the set of the Volterra kernels for the nonlinear system. The zeroth order Volterra functional is given by $G_0\left[g_0, x(t)\right] = g_0$; the first and second order functionals are given by

$$G_1\left[g_1, x(t)\right] = \int_{\mathbb{R}} g_1(\tau)x(t-\tau)d\tau$$

$$G_2\left[g_2, x(t)\right] = \int_{\mathbb{R}^2} g_2(\tau_1,\tau_2)x(t-\tau_1)x(t-\tau_2)d\tau_1 d\tau_2$$

The first order Volterra functional $G_1\left[g_1, x(t)\right]$ is the linear convolutional integral, and its kernel $g_1(t)$ is the most familiar time-domain description of an LTI system (*i.e.* impulse response function). In the following section, we present a mathematical framework to apply Volterra series expansion to a three layered MLP estimating the posterior probabilities of phonemes.

3.1. Volterra Kernel Identification: Three layered MLP

Fig. 1(b) shows a part of the phoneme posterior estimator that is modeled using Volterra series. It is a multi-input, multi-output, nonlinear time-invariant system comprising of an LTI filter bank followed by the MLP. The input to the system are log energies from the auditory analysis $x_k(t),\ k = 1, 2 \ldots K$, where K is the number of auditory channels. The output of the system is the accumulated sum $y^j(t),\ j = 1, 2 \ldots N$ before the output nonlinearity, where N is the number of output nodes in the MLP. The Volterra series expansion of such a system can be expressed as

$$y^j\left(t\right) = g_0^j \ + \ \sum_{k_1=1}^{K} \int_{\tau_1} g_{k_1}^j\left(\tau_1\right) x_{k_1}\left(t-\tau_1\right) d\tau_1 \ + \ \sum_{k_1=1}^{K}\sum_{k_2=1}^{K}$$

$$\int_{\tau_1}\int_{\tau_2} g_{k_1 k_2}^j\left(\tau_1,\tau_2\right) x_{k_1}\left(t-\tau_1\right) x_{k_2}\left(t-\tau_2\right) d\tau_1 d\tau_2 + \ldots \quad (1)$$

where, the terms $g_0^j,\ g_{k_1}^j\left(\tau_1\right),\ g_{k_1 k_2}^j\left(\tau_1,\tau_2\right)$ are the zeroth, first, and second order Volterra kernels respectively of the phoneme j. The variables $\tau_1, \tau_2 \ldots$ denote time, and $k_1, k_2 \ldots$ denotes the frequency on the Mel or Bark scale. We identify the above Volterra kernels in terms of the impulse response of the LTI system and the parameters of the MLP.

Even though the above system is a discrete-time system, we use continuous-time notations through out this paper for clarity. The LTI system, which is a part of feature extraction consists of a bank of L FIR filters, each with an impulse response of $h_l(t),\ l = 1, 2 \ldots L$. The component of the feature vector $u_{k,l}\left(t\right)$ is obtained by convolving the input $x_k(t)$ with the impulse response $h_l(t)$, and given by

$$u_{k,l}\left(t\right) = \int_{\tau} h_l\left(\tau\right) x_k\left(t-\tau\right) d\tau. \quad (2)$$

The MLP consists of $K \times L$ input nodes which is same as the dimension of the feature vector, M hidden nodes, and N output nodes. The input $s_i\left(t\right)$ to the hidden nonlinearity function $\phi_i(.)$ is the linear combination of the input features $u_{k,l}(t)$ weighted by the MLP weights from the input to the hidden layer $w^i_{k,l}$, and given by

$$s_i\left(t\right) = \sum_{k=1}^{K}\sum_{l=1}^{L} w_{k,l}^i u_{k,l}\left(t\right). \quad (3)$$

Here, we assume that features presented to the MLP are not normalized. Kernel identification for normalized features is discussed in section 3.2. The accumulated sum at the j^{th} output node is the linear combination of the outputs at the hidden layer and the weights connecting the hidden and the output layer of the MLP, and given by

$$y^j\left(t\right) = \sum_{i=1}^{M} c_i^j \ \phi_i\left(s_i\left(t\right)\right). \quad (4)$$

$\phi_i(.) = \phi(h_i + .)$ is the nonlinearity at the i^{th} hidden node, where h_i is the bias and $\phi(.)$ is the nonlinear activation function (sigmoid, hyperbolic tangent). To derive the Volterra kernels, $\phi_i(.)$ is approximated using a polynomial expansion of the form

$$\phi_i\left(s_i\left(t\right)\right) = a_{0,i} + a_{1,i}s_i\left(t\right) + a_{2,i}s_i\left(t\right)^2 + \ldots, \quad (5)$$

where the coefficients $a_{0,i}, a_{1,i} \ldots$ are scalar constants. Polynomial expansion of the nonlinearity and the estimation of the coefficients is discussed in section 3.3. By substituting (5) in (4), we obtain

$$y^j\left(t\right) = \sum_{i=1}^{M} c_i^j \ \left[a_{0,i} + a_{1,i}\ s_i\left(t\right) + a_{2,i}\ s_i\left(t\right)^2 + \ldots\right]. \quad (6)$$

By substituting (2) and (3) in (6), and comparing the resulting equation to the Volterra series expansion in (1), we are able to identify the Volterra kernels. The first three Volterra kernels are given by

$$g_0^j = \sum_{i=1}^{M} c_i^j \; a_{0,i} \tag{7}$$

$$g_{k_1}^j(\tau_1) = \sum_{i=1}^{M} c_i^j \; a_{1,i} \sum_{l_1=1}^{L} w_{k_1 l_1}^i h_{l_1}(\tau_1) \tag{8}$$

$$g_{k_1 k_2}^j(\tau_1,\tau_2) = \sum_{i=1}^{M} c_i^j a_{2,i} \sum_{l_1=1}^{L} \sum_{l_2=1}^{L} w_{k_1 l_1}^i w_{k_2 l_2}^i h_{l_1}(\tau_1) \, h_{l_2}(\tau_2) \tag{9}$$

The intermediate steps in the derivation of the Volterra kernels are described in [9]. The identified Volterra kernels are functions of the impulse responses of the filters in the LTI system, and the parameters of the functions are determined by the weights of the MLP.

3.2. Volterra Kernel Identification: Feature Normalization

In practice, the input features to the MLP are normalized to zero mean and unit variance so that the operating point on the hidden activation function is in the linear region, leading to a faster convergence of the back propagation training algorithm [10]. Suppose that the feature vector component $u_{k,l}(t)$ has a mean $\mu_{k,l}$ and a standard deviation $\sigma_{k,l}$. By substituting the normalized feature component $\hat{u}_{k,l}(t) = (u_{k,l}(t) - \mu_{k,l})/\sigma_{k,l}$ in (3), we obtain

$$s_i(t) = \sum_{k=1}^{K} \sum_{l=1}^{L} w_{k,l}^i \, \hat{u}_{k,l}(t) \; = \sum_{k=1}^{K} \sum_{l=1}^{L} \hat{w}_{k,l}^i \; u_{k,l}(t) - \Delta_i \tag{10}$$

$$where, \quad \hat{w}_{k,l}^i = \frac{w_{k,l}^i}{\sigma_{k,l}}, \quad and \quad \Delta_i = \sum_{k=1}^{K} \sum_{l=1}^{L} w_{k,l}^i \frac{\mu_{k,l}}{\sigma_{k,l}} \tag{11}$$

The parameter Δ_i can be interpreted as the correction to the hidden bias introduced by the feature mean. Volterra kernels can be derived in the same way as described in Section 3.1, but using (10) instead of (3). The first two Volterra kernels are identified as

$$g_0^j = \sum_{i=1}^{M} c_i^j \hat{a}_{0,i} \;\; ; \quad g_{k_1}^j(\tau_1) = \sum_{i=1}^{M} c_i^j \; \hat{a}_{1,i} \sum_{l_1=1}^{L} \hat{w}_{k_1 l_1}^i h_{l_1}(\tau_1) \tag{12}$$

All higher order Volterra kernels are of the same mathematical form as those corresponding to unnormalized features, but the weights and the coefficients of polynomial expansion are appropriately modified by the mean and variance of the features. The new weights connecting the input and hidden layer of the MLP $\hat{w}_{k,l}^i$ is given by (11). The new polynomial coefficients are weighted linear combination of the coefficients of the polynomial expansion (5). The new r^{th} order polynomial coefficient at i^{th} hidden node is given by

$$\hat{a}_{r,i} = \sum_{n=r}^{\infty} \binom{n}{r} a_{n,i} \quad (-\Delta_i)^{n-r} \tag{13}$$

It can be seen from (12) and (13) that due to feature normalization, the Volterra kernels are also in the form of an infinite series. However, in practice the order of the Volterra series as well as the Volterra kernels is decided by the order of the polynomial approximation of the hidden nonlinearity. Moreover, the coefficients $a_{n,i}$ in (13) approach towards zero as the order n is increased. In cases where the features are zero mean but non-unit variance, the polynomial coefficients remain unchanged as $\Delta_i = 0$, but MLP weights are appropriately scaled by the feature variance.

3.3. Polynomial expansion of the activation function

A key aspect in the derivation of the Volterra kernels is the polynomial expansion (5) of the nonlinearity at the hidden nodes. Polynomial expansion of activation functions such as sigmoid is divergent if approximated for all possible values of the input $(-\infty, \infty)$. However, as a consequence of feature normalization, the operating point on the nonlinearity is in a relatively small region containing the linear part of the function.

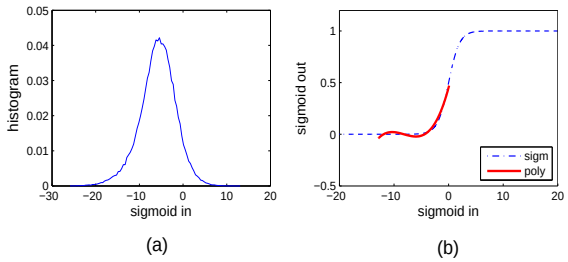

Fig. 2. *(a) Histogram of the input to the sigmoid at an hidden node. (b) Sigmoid function and its 3^{rd} order polynomial expansion.*

Fig. 2(a) shows the histogram of the input (which includes the bias) to the sigmoid function at a hidden node, and is obtained on the cross-validation data. We fit a polynomial function of certain order in the range of values observed in the histogram, leaving out a small fraction on the tail. The coefficients of the polynomial are optimized to minimize the least mean square error between the sigmoid function and its polynomial approximation in the region of interest. Fig. 2(b) is the plot of the sigmoid activation function and its polynomial approximation. Since the hidden bias is incorporated in the polynomial expansion, the estimated coefficients are different for each hidden node.

3.4. Interpretation of Volterra kernels

The first order Volterra kernel $g_k^j(t)$ (t denotes time, k denotes frequency, and j denotes the phoneme) is the linear transfer function of the posterior feature extraction system. The time-reversed linear kernel can be interpreted as a matched filter capturing the spectro-temporal patterns learned by the system. The second order kernel $g_{k_1 k_2}^j(t_1, t_2)$ for the phoneme j reveals the correlations across different frequency bands (k_1, k_2) at different times (t_1, t_2). Similarly the higher order Volterra kernels reveal the higher order correlations in the nonlinear system.

4. VOLTERRA ANALYSIS ON MFB FEATURES

To demonstrate the applicability of Volterra series expansion, we analyze a posterior feature extraction system, where the MLP trained on the standard TIMIT database using using Mel filter bank energy (MFB) features. The log-energies from the 26 auditory channels are presented to the MLP with a context of 170 ms. Hence the LTI system in Fig. 1 is a bank of 17 FIR filters with shifted Kronecker delta impulse response functions. The input layer of the MLP consists of 442 nodes, the hidden layer consists of 1000 nodes, and the output layer consists of 39 nodes corresponding to the number of phonemes. The training set consists of 153 minutes (375 speakers), cross-validation set consists of 34 minutes (87 speakers), and test set consists of 68 minutes (168 speakers) of speech.

The Volterra kernels are derived using (12). We fit a polynomial function of order 3 to the hidden nonlinearity, leaving out 5% of the points on the tail of the histogram. The identified kernels are applied in the Volterra series (1) to estimate the phoneme posterior probabilities. The estimated probabilities are evaluated by applying them in isolated phoneme recognition experiments [1]. Viterbi algorithm is applied on the phoneme posterior probabilities with a minimum duration of three states per phoneme [3]. Table 1 shows the phoneme classification accuracy obtained by using linear and quadratic approximation of the MLP using Volterra series. The accuracy obtained using Volterra series should converge to the accuracy obtained using the MLP as the order of the series is increased.

model	series order	accuracy (%)
linear	1	38.2
quadratic	2	43.7
MLP	∞	77.9

Table 1. *Phoneme classification accuracy obtained by linear and quadratic approximation of the MLP using Volterra series.*

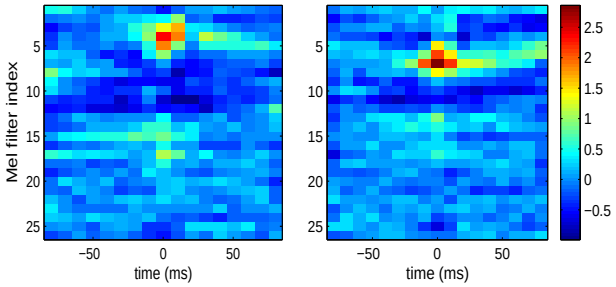

Fig. 3. *Linear Volterra kernels for phonemes /iy/ (left) and /eh/ (right)*

Fig. 3 shows the first order Volterra kernel for phonemes /iy/ (*e.g.* b**ea**t) and /eh/ (*e.g.* b**e**t). It can be seen that in the case of phoneme /iy/, the system has learned to emphasize 200-300 Hz frequency band which corresponds to its first formant. In case of /eh/, the system has learned to emphasize slightly higher frequency region of 400-500 Hz, which corresponds to its first formant. Fig. 4 shows the first order Volterra kernel for the phonemes /s/ (*e.g.* **s**ee) and /z/ (*e.g.* **z**oo). It can be seen that for both these phonemes, the system has learned to emphasize the higher frequency regions. However, the unvoiced phoneme /s/ is distinguished from the voiced phoneme /z/ by the lack of energy in its low frequency region.

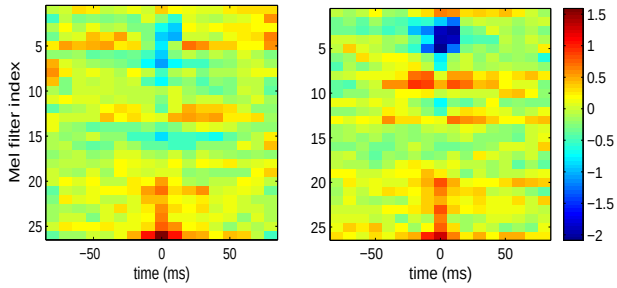

Fig. 4. *Linear Volterra kernels for phonemes /z/ (left) and /s/ (right)*

[1]Phoneme classification facilitates accurate analysis of the results as insertions and deletions are avoided. However, the trends observed in phoneme classification are also observed in phoneme recognition experiments.

5. SUMMARY AND CONCLUSION

The main objective of this work was to provide a framework to apply Volterra series to analyze MLP based phoneme posterior probability estimation. We include a part of the feature extraction (LTI system following the auditory analysis) in the analysis framework because the Volterra kernels can be interpreted as spectro-temporal patterns.

The applicability of Volterra series is demonstrated by analyzing an MLP trained using MFB features. However, the proposed framework is generic and can be applied where the MLP is preceded by an LTI system. Volterra analysis for MRASTA features [8] is straight forward and is shown in [9]. In case of MFCC, application of Volterra series is not straight forward as the DCT transform mixes the energies across different auditory channels. However, the cosine transformation can be incorporated into the weights of the MLP and the proposed framework can be applied as discussed in [9].

In this work, the linear Volterra kernels are interpreted as spectro-temporal patterns. The second order kernels could reveal useful correlations across different frequency channels at different time instants. The spectro-temporal patterns given by the Volterra kernels may not be consistent with the existing acoustic phonetic knowledge of phonemes in all aspects. This is because the Volterra kernels can only reveal the information learned by the MLP to discriminate among phonemes.

Analytical identification of Volterra kernels becomes complicated if the phoneme posterior estimator is more complex such as an MLP with more than one hidden layer or a hierarchal structure of more than one MLP. In such a case, the system can be modeled using Wiener series [11], whose functionals are orthogonal with respect to white Gaussian noise. The Wiener kernels are estimated using cross-correlation based methods [5], and Volterra kernels can be subsequently computed from the Wiener kernels.

6. REFERENCES

[1] N. Morgan et al., "Pushing the Envelope - Aside," *IEEE Signal Processing Magazine*, vol. 22, no. 5, pp. 81–88, 2005.

[2] Q. Zhu, B. Chen, N. Morgan, and A. Stolcke, "On Using MLP Features in LVCSR," *Proc. of Interspeech*, pp. 921–924, 2004.

[3] H. Bourlard and N. Morgan, *Connectionist Speech Recognition – A Hybrid Approach*, Kluwer Academic Publishers, 1994.

[4] H. Hermansky, D.P.W. Ellis, and S. Sharma, "Tandem Connectionist Feature Extraction for Conventional HMM Systems," *Proc. of ICASSP*, pp. 1635–1638, 2000.

[5] Y.W. Lee and M. Schetzen, "Measurement of Wiener Kernels of a Non-linear System by Cross-correlation," *International Journal of Control*, vol. 2, pp. 237–254, 1965.

[6] V. Volterra, *Theory of Functunals and of Integro-Differential Equations*, Dover, New York, 1930.

[7] G. Stegmayer, "Volterra Series and Neural Networks to model an Electronic Device Nonlinear Behavior," *Proc. of IEEE Conf. Neural Networks*, vol. 4, pp. 2907–2910, 2004.

[8] H. Hermansky and P. Fousek, "Multi-Resolution RASTA Filtering for Tandem based ASR," *Proc. of Interspeech*, 2005.

[9] J. Pinto et al., "Volterra Series for Analyzing MLP Based Phoneme Posterior Estimator," *Idiap Research Report*, , no. 69, 2008.

[10] Y. LeCun, L. Bottou, G.B. Orr, and K.-R Muller, "Effecient BackProp," *Neural Networks: Tricks of the Trade*, 1998.

[11] N. Wiener, *Nonlinear Problems in Random Theory*, MIT Press, 1966.