Learning Waveform-Based Acoustic Models using Deep Variational Convolutional Neural Networks

Dino Oglic, Zoran Cvetkovic, and Peter Sollich

***Abstract*—We investigate the potential of stochastic neural networks for learning effective waveform-based acoustic models. The waveform-based setting, inherent to fully end-to-end speech recognition systems, is motivated by several comparative studies of automatic and human speech recognition that associate standard non-adaptive feature extraction techniques with information loss, which can adversely affect robustness. Stochastic neural networks, on the other hand, are a class of models capable of incorporating rich regularization mechanisms into the learning process. We consider a deep convolutional neural network that first decomposes speech into frequency sub-bands via an adaptive parametric convolutional block where filters are specified by cosine modulations of compactly supported windows. The network then employs standard non-parametric 1D convolutions to extract relevant spectro-temporal patterns while gradually compressing the structured high dimensional representation generated by the parametric block. We rely on a probabilistic parametrization of the proposed neural architecture and learn the model using stochastic variational inference. This requires evaluation of an analytically intractable integral defining the Kullback–Leibler divergence term responsible for regularization, for which we propose an effective approximation based on the Gauss–Hermite quadrature. Our empirical results demonstrate a superior performance of the proposed approach over comparable waveform-based baselines and indicate that it could lead to robustness. Moreover, the approach outperforms a recently proposed deep convolutional neural network for learning of robust acoustic models with standard FBANK features.**

***Index Terms*—Convolutional neural networks, parametric filters, variational inference, waveform-based speech recognition.**

I. Introduction

Automatic speech recognition systems typically operate in low-dimensional feature spaces designed to achieve invariances inherent to speech production and human speech recognition [1–3]. Log Mel-filter bank values (FBANK) and their de-correlated variant known as Mel-frequency cepstral coefficients (MFCC) are two of the most frequently used feature extraction techniques of this kind [4, 5]. Several comparative studies of automatic and human speech recognition [6–8] suggest that the information loss inherent to such feature extraction techniques can adversely affect robustness to standard environmental distortions arising from additive and channel (linear filtering) noise [9, 10]. Motivated by this, we propose an effective and principled approach for learning of robust acoustic models in the waveform domain. A difficulty in the waveform setting is the sheer size of the training data required for learning effective waveform-based models. More specifically, the requirement for more than $2,000$ hours of speech in [11, 12] translates into weeks of training on a typical device with GPU support. Our aim is to tackle this problem by incorporating relevant inductive bias into the learning process and allow for learning of effective waveform-based acoustic models using moderately sized datasets. There are two components in our approach, one dealing with the design of neural architectures and the other with learning of the corresponding parameters.

Section II is concerned with the design of neural architecture, which should perform automatic feature extraction by avoiding fast compression schemes associated with information loss when operating with standard non-adaptive filterbank features [6–8]. We design the neural network as a Lipschitz continuous operator that maps speech waveform frames into a feature space in such a way that small perturbations in the inputs caused by local translations and diffeomorphisms result in relatively small changes in the pre-softmax network outputs. As we operate in the waveform domain, the first layer of our convolutional network extracts information relevant for discrimination between phonetic units by decomposing a speech frame into frequency sub-bands using a set of parametric band-pass filters. The filters are defined by cosine modulations of compactly supported windows and allow for embedding of waveform signals into a structured high-dimensional space where we hypothesize that phonetic units will be easier to separate. The network then employs standard 1D convolutional layers with non-parametric filters for extraction of relevant spectro-temporal patterns while gradually compressing the structured representation generated by the sub-band decomposition. The outputs of the last such convolutional block are passed to a multi-layer perceptron (MLP) with a softmax output.

The learning component of our approach is described in Section III. We propose to learn a probabilistic parametrization of our architecture using variational inference. The motivation for this comes from the fact that for robustness one needs to be able to select the operator mapping with a good Lipschitz constant. The role of probabilistic parametrization and variational inference is to regularize the training process, thus allowing us to learn a robust feature representation of speech signals. This is different from a typical acoustic model, which employs an artificial neural network with real-valued parameters. Such a *deterministic* parametrization of the network fails to capture the uncertainty of individual parameters and their importance for the learning task. Bayesian machine learn-

D. Oglic and Z. Cvetkovic are with the Department of Engineering, King's College London. Correspondence to: dino.oglic@uni-bonn.de.

P. Sollich is with the Department of Mathematics, King's College London, and the Institute for Theoretical Physics, University of Göttingen.

2329-9290 © 2021 IEEE. Citation information: DOI 10.1109/TASLP.2021.3104193, IEEE/ACM Transactions on Audio, Speech, and Language Processing.

ing provides a principled framework for modeling uncertainty by finding plausible models that could explain the observed data [13, 14]. In particular, a (deterministic) neural network with fixed parameter values models the conditional probability of a sub-phonetic unit given a speech frame. In *stochastic* neural networks one additionally assumes that the parameters follow some prior distribution. The latter coupled with the aforementioned likelihood gives rise to a posterior distribution of parameter values conditioned on the observed data. Such posteriors are typically defined via analytically intractable integrals that can be approximated using scalable inference techniques such as stochastic variational inference [15–17]. In particular, the main idea is to approximate intractable posteriors by optimizing over parameters of an a priori selected family of variational distributions. The optimization objective in variational inference consists of two terms: *i*) the expected negative log-likelihood of the model, where the expectation is taken with respect to the variational distribution, and *ii*) the Kullback–Leibler divergence that performs regularization. The expectation in the first term is approximated by sampling the variational distribution, which is typically given by a Gaussian mean field. In this way, the variational formulation injects randomness into the forward pass that computes the loss associated with a particular mini-batch. As a result, stochastic neural networks can capture parameter uncertainty and are less sensitive to perturbations in parameter values, as well as less susceptible to over-fitting [15, 17]. A further regularization effect, incorporated via the Kullback–Leibler divergence, is specified by an analytically intractable integral. For this we propose an effective approximation based on the Gauss–Hermite quadrature. Variational inference has been used previously in speech recognition, albeit in a different context, to maintain the balance between a dataset size and model complexity [18, 19]. In addition to this, a high correlation between the uncertainty in individual parameters and their importance for speech recognition has been observed in stochastic recurrent nets [20, 17]. Previous work, however, does not operate in the waveform domain, focuses on recurrent nets and considers variational inference separately from the properties encoded into the architecture (i.e. Lipschitz continuity in our case).

In Section IV, we focus on the relationship with prior work on speech recognition in the waveform domain. We then evaluate the proposed approach empirically on three benchmark datasets for automatic speech recognition: TIMIT, AURORA4, and AMI-IHM. A summary of our empirical results is provided in Section V. The ablation study (evaluating the effectiveness of individual components in our approach) demonstrates that acoustic models based on modulation filter learning can be more effective, in a statistically significant way, than the ones with non-adaptive filters. Moreover, the experiments indicate that the proposed approximation scheme based on the Gauss–Hermite quadrature provides a general (with respect to the choice of prior function) and effective means for approximating the Kullback–Leibler divergence term. The experiments on the TIMIT dataset demonstrate that the approach does not over-fit despite using a rather large network on what in speech recognition is considered to be a small dataset. Moreover, our results on AURORA4 show that the approach is capable of learning a noise robust model, outperforming significantly the state-of-the-art baselines for waveform-based speech recognition on this dataset. It is also promising that on the same dataset the approach outperforms a recently proposed deep convolutional network for learning of robust acoustic models with standard FBANK features [21]. The experiments on AMI (conversational speech, without i-vectors or data augmentation) show that the approach outperforms recently proposed architectures for raw speech (see [22] and [23]) and performs on par with a state-of-the-art FBANK/MFCC based deep time-delay neural network (TDNN) model [24]. Thus, our empirical contributions provide comprehensive evidence for the effectiveness of variational neural networks operating directly in the waveform domain.

II. PARZNETS — DEEP CONVOLUTIONAL NEURAL NETWORKS FOR WAVEFORM-BASED SPEECH RECOGNITION

This section describes an artificial neural network for learning acoustic models in the waveform domain. We first provide a brief overview of the relevant building blocks of the architecture (Section II-A) and then introduce a parametric convolutional layer responsible for decomposition of speech signals into frequency sub-bands (Section II-B). The section concludes with a theoretical analysis demonstrating that the proposed neural architecture defines a Lipschitz continuous operator in the waveform domain (Section II-C).

A. Overview of the Neural Architecture

We would like to design an architecture capable of embedding redundancies into the representation, thereby avoiding significant overlaps between positioning of different phonetic units while allowing for a fair amount of additive noise and distortion at inputs. Motivated by this, we extract information relevant for discrimination between phonetic units via a parametric Parzen convolutional block (Section II-B) that decomposes a waveform frame into frequency sub-bands, thereby embedding the signal into a high-dimensional space of high-resolution spectro-temporal patterns (illustrated in Fig. 1, PARZNETS 1D). A notable difference compared to non-adaptive feature extraction operators (FBANK and MFCC) is the use of a RELU activation function instead of the modulus (squared) non-linearity. Mallat [25] has demonstrated that this change in activation function does not affect the theoretical properties of such operators. Moreover, it has been established recently that neural networks with RELU activations realize piecewise linear functions and we therefore use that non-linearity throughout the network [26]. The main motivation behind this choice is to avoid further confounding effects between signal and noise that might otherwise arise from additional sources of non-linearity in the automatic feature extraction process (it is well known, for example, that the effects of channel noise can be amplified by non-linearities). To extract relevant patterns from such a sub-band decomposition/representation, we rely on standard non-parametric convolutional filters and pass the Parzen sub-bands to double convolutional blocks with 5 sample long filters (see CONV-CONV in Fig. 1). The gradual compression of the spectro-temporal representation is achieved by applying the max

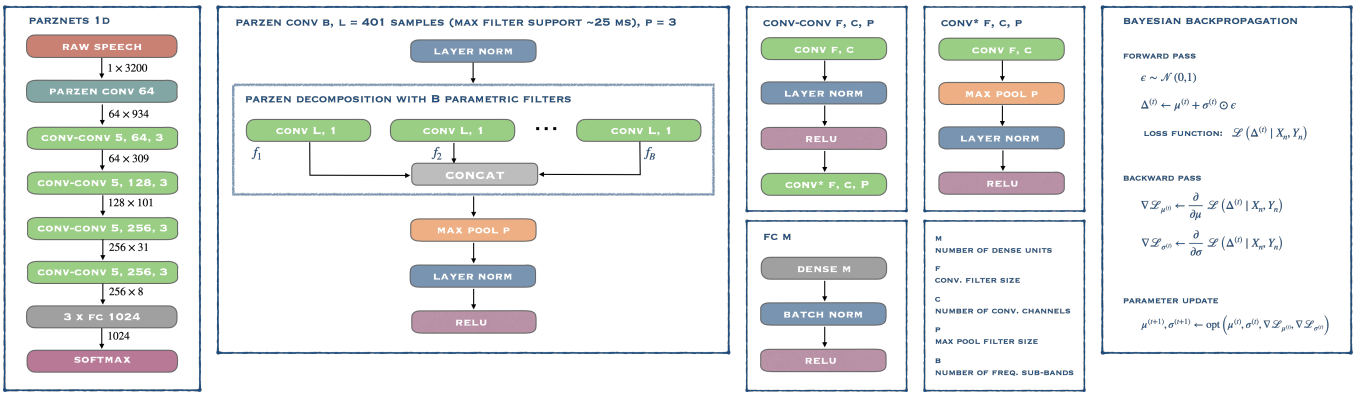

Fig. 1: This is a schematic illustrating PARZNETS with 1D convolutional operators. The illustration is supplemented with the Parzen block (the second panel on the left) that decomposes a raw speech frame into frequency sub-bands and a pseudo-code description of Bayesian backpropagation used in variational inference. The loss function in the rightmost panel refers to the variational objective that is described in detail in Section III.

pooling operator with size 3 (after each pair of non-parametric convolutional blocks). Previous work [27] has demonstrated that a composition of convolution with max pooling tends to provide approximate local time-translation invariance. In our preliminary experiments, we investigated the effectiveness of $\max$ and (weighted) ℓ_p average pooling operators, and observed that the former works the best in combination with RELU activations. The features extracted by the last convolutional block are passed to an MLP block with three hidden layers (i.e., fully connected layers denoted by FC in Fig. 1), followed by a softmax output block.

B. Parzen Block for Sub-band Decomposition of Speech Signals

It has been demonstrated recently that feature extraction operators that combine band-pass filtering with the modulus (square) non-linearity and (weighted) local averaging are approximately locally translation invariant and Lipschitz continuous [28]. A potential shortcoming of these operators is the fact that filter parameters are selected a priori without relying on data. As a result, the hypothesis space is selected beforehand and does not necessarily provide an ideal inductive bias for all learning tasks. Moreover, the power spectral averaging that is characteristic of these operators is typically performed over speech segments of 25 or 32 ms [28, 29], which could be compressing the relevant information too fast into the resulting features. As a result of such compression, the feature extraction operator might be discarding the information relevant for robustness. Motivated by this, we have designed the Parzen convolutional block to tackle these shortcomings. In particular, the block does not rely on a priori selected filters but learns these via parametric convolutions that have a strongly encoded inductive bias. Moreover, the adaptive Parzen convolutional block embeds a waveform frame into a structured high dimensional space rather than compressing it into a small number of features. The latter is an important difference compared to MFCC and FBANK coefficients, which do not focus on embedding redundancies into the representation. As explained above, the Parzen sub-band decomposition is followed by a gradual compression of the representation using a combination of convolutional and max pooling operators.

In speech recognition, band-pass filtering of signals is traditionally performed by (weighted) averaging of power spectra [see 5, 30] computed over speech frames of fixed duration. Alternatively, the signal can be convolved with a filter directly in the time domain. To that end, we consider a family of differentiable band-pass filters based on cosine modulations of compactly supported Parzen windows [31]. In particular, we employ the squared Epanechnikov window function given by

$$k_\gamma(t) = \max\left\{0, 1 - \gamma t^2\right\}^2 \quad , \tag{1}$$

where γ is a parameter controlling the window width, and implicitly its frequency bandwidth. The filter can be made more frequency selective by increasing its exponent (illustrated above with the square operator), which is a consequence of increasing its order of differentiability. To allow for flexible placement of the center frequency we rely on cosine modulation. Thus, Parzen filters are defined with only two differentiable parameters, η controlling the modulation frequency and γ controlling the filter bandwidth:

$$\phi_{\eta,\gamma}(t) = \cos(2\pi\eta t) \cdot k_\gamma(t) \quad . \tag{2}$$

As illustrated in Fig. 1 (the leftmost panel), for each filter configuration $\{(\eta_i, \gamma_i)\}_{i=1}^B$, we use Eq. (2) to generate a one-dimensional convolutional filter with maximum length given by the number of samples in 25 ms of speech; filters with shorter support are symmetrically padded with zeros. The outputs of parametric convolutions are concatenated into a high dimensional spectro-temporal decomposition of a signal and then passed to a max pooling operator, followed by layer normalization [32]. As all of the operations in this parametric block are differentiable, it is possible to construct an auto-differentiation graph that seamlessly provides gradients with respect to parameters of Parzen filters. In comparison to wavelet filters [33], the Parzen convolutional block offers additional flexibility by allowing independent control over bandwidth and modulation frequency. Moreover, the block optimizes for the positioning of the two parameters while having the parametric form of the filter factored into the optimization. This can be seen as a more flexible approach compared also to the two-step procedure employed by [23], where filter cut-off frequencies are optimized with respect to a fixed-length rectangular window, and then a Hamming window is superimposed to suppress the ripple effects.

C. Lipschitz Continuity of the Operator Mapping

We start with a review of Lipschitz continuity for operator mappings and properties relevant for their robustness. Following this, we demonstrate that the principle for the design of neural architectures outlined in Section II-A and Fig. 1 defines a Lipschitz continuous operator in the waveform domain.

Let $\mathcal{L}(\mathbb{R})$ denote the space of square integrable functions defined on $\mathbb{R}$ and consider a continuous signal $f \in \mathcal{L}(\mathbb{R})$. An operator $\Phi\colon \mathcal{L}(\mathbb{R}) \to \mathcal{H}$ is a mapping of a signal into a Hilbert space $\mathcal{H}$. Let $T_c f(t) = f(t-c)$ denote the translation of a signal f by some constant $c \in \mathbb{R}$. An operator Φ is called *translation invariant* if $\Phi(T_c f) = \Phi(f)$ for all $f \in \mathcal{L}(\mathbb{R})$ and $c \in \mathbb{R}$. The spectrogram of a signal is an operator designed to capture variations in the power spectrum over time. It can provide an approximately locally time-translation invariant representation over durations limited by a window [28]. While the spectrogram of a signal can provide local time-translation invariance, Mallat [29] has demonstrated that it does not necessarily provide stability to the action of a small diffeomorphism (e.g., speed perturbation of an utterance). Let $D_\tau\colon \mathcal{L}(\mathbb{R}) \to \mathcal{L}(\mathbb{R})$ be a diffeomorphism of a signal (i.e., invertible function that maps one differentiable manifold to another such that both the function and its inverse are smooth) given by $D_\tau f(t) = f(t - \tau(t))$, where $\tau(t) \in \mathcal{C}^2(\mathbb{R})$ is a displacement field and $\mathcal{C}^2(\mathbb{R})$ denotes the space of twice continuously differentiable functions over the reals. For example, one can take $\tau(t) = \epsilon t$ with $\epsilon \in \mathbb{R}$ and $\epsilon \to 0$. To preserve stability relative to a small diffeomorphism of a signal, it is sufficient to ensure that the operator Φ is Lipschitz continuous [29, 28]. A translation invariant operator Φ is Lipschitz continuous with respect to actions of $\mathcal{C}^2$-diffeomorphisms if for any compact $\Omega \subset \mathbb{R}$ there exists a constant L such that for all signals $f \in \mathcal{L}(\mathbb{R})$ supported on Ω and all $\tau \in \mathcal{C}^2(\mathbb{R})$ it holds that [for more details see, e.g., 29]

$$\begin{aligned}&\|\Phi(f) - \Phi(D_\tau f)\|_{\mathcal{H}} \leq L \|\mathbb{I} - D_\tau\|_\infty \|f\| \\ &:= L\left(\sup_{t\in\Omega} \|\nabla\tau(t)\| + \sup_{t\in\Omega} \|\nabla\nabla\tau(t)\|\right)\|f\| \ ,\end{aligned}$$

where $\mathbb{I}$ denotes the identity mapping. The Lipschitz continuity of operator Φ implies invariance to *local translations* and/or signal warping by a diffeomorphism $\tau(t)$, up to the first and second order deformation terms [29]. Such signal perturbations typically come as a result of variability in speech production and differences between speakers. Another aspect of robust representations is the ability to withstand a fair amount of additive and channel/linear noise. It is easy to show (e.g., using the convolution theorem) that such a perturbation of a clean speech signal amounts to a linear transformation of its representation in the frequency domain. Thus, an operator that is Lipschitz continuous over the sub-band decomposition of a signal has the potential to work effectively on noisy speech. In particular, a noise corrupted signal is a linear transformation of the clean signal in the frequency domain and will be contained within a ball of constant radius centered at the clean signal. An operator that is Lipschitz continuous over the frequency representation of a signal will exhibit small variations over such balls and, thus, it can provide stability relative to additive and channel noise. It is, however, important to point out that the robustness of such an operator quantitatively depends on the value of the Lipschitz constant.

The operator defined by our neural network maps a frame of raw speech $x \in \mathbb{R}^d$ into a vector of pre-softmax outputs $z \in \mathbb{R}^s$, where d is the number of samples in the input frame and s is the dimension of the pre-softmax representation. Moreover, this is achieved by having an intermediate representation of the signal in the frequency domain via sub-band decomposition performed by the Parzen block. The operator mapping can be expressed as a composition of functions

$$\Phi(x) = \left(\rho_l \circ \rho_{l-1} \circ \dots \rho_1\right)(x) \ ,$$

where ρ_i represents the RELU activation function, linear or pooling operator. In particular, the building blocks of our architecture are fully connected and convolutional layers, which are both linear operators and can be realized as matrix-vector multiplications [see, e.g., 34]. For a fully connected block with weights W and bias b, the Lipschitz constant L is given by

$$\|Wz + b - Wz' - b\|_2 \leq L \|z - z'\|_2 .$$

Thus, the minimal value of the Lipschitz constant is equal to $L = \sup_{z\in\mathcal{B}} {}^{\|Wz\|_2}/_{\|z\|_2}$, where $\mathcal{B}$ is a ball of constant radius containing all the layer inputs in its interior. The convolution blocks can also be realized via matrix-vector multiplications using doubly block circulant matrices [34]. Thus, a good Lipschitz constant can be obtained by keeping low the upper bounds on the weights in linear blocks and convolutional filters, while at the same time optimizing for the operator mapping such that the sub-phonetic units are linearly separable.

Gouk et al. [34] have demonstrated that the RELU activation function is Lipschitz continuous with constant one. This activation function is also monotonic and, thus, defines a contraction. The same holds for the max operator used for signal pooling, as demonstrated with the following proposition.

Proposition 1. *The max pooling operator is a Lipschitz continuous function with constant one.*

Proof. The max pooling operator can be expressed as $\iota(z) = \max_{1\leq j\leq k} \sigma_{i,j}(z)$, where $\sigma_{i,j}(z)$ is the j-th output of the i-th network layer that takes a vector z as input. We will show that

$$\left|\max_{1\leq j\leq k} \sigma_{i,j}(z) - \max_{1\leq j\leq k} \sigma_{i,j}(z')\right| \leq \|\sigma_i(z) - \sigma_i(z')\|_2 \ .$$

We can, without loss of generality, assume that $\iota(z) \geq \iota(z')$. Denote $j_0 = \arg\max_{1\leq j\leq k} \sigma_{i,j}(z)$. Then,

$$\begin{aligned}&|\iota(z) - \iota(z')| = \max_{1\leq j\leq k} \sigma_{i,j}(z) - \max_{1\leq j\leq k} \sigma_{i,j}(z') \leq \\ &\sigma_{i,j_0}(z) - \sigma_{i,j_0}(z') \leq \max_{1\leq j\leq k} (\sigma_{i,j}(z) - \sigma_{i,j}(z')) \leq \\ &\|\sigma_i(z) - \sigma_i(z')\|_\infty \leq \|\sigma_i(z) - \sigma_i(z')\|_2 \ .\end{aligned} \tag{3}$$

□

As the proposed neural architecture is defined using a composition of Lipschitz continuous functions, the resulting operator mapping is also Lipschitz continuous. To ensure that the training procedure selects a good Lipschitz constant, we propose to use a probabilistic parametrization for our network and

learn the corresponding parameters using stochastic variational inference, as described in the next section.

While Lipschitz continuity of neural architectures has already been associated with robust representation learning [e.g., see 34], this is the first work that provides an explanation for possible advantages of the filterbank over sample-based audio processing. In particular, in order to learn an effective (relative to longer time-shifts, additive and channel/linear noise) waveform-based representation of speech signals one can design the neural architecture as a Lipschitz continuous operator in the waveform-domain, with an intermediate representation in the frequency domain that can be realized using a sub-band decomposition of the signal (the Parzen block in our case).

III. Learning Parznets using Stochastic Variational Inference

In deterministic neural networks, parameters/weights are real-valued and one performs inference by optimizing a loss function over them. Performing inference in stochastic/probabilistic neural networks, on the other hand, requires a posterior distribution over parameters given data [17]. For a fixed setting of weights, a deterministic neural network with *softmax* outputs models the conditional probability of a categorical label $y \in \mathcal{Y}$ given an instance $x \in \mathcal{X}$ using an exponential family model [e.g., see 35, 36]. In stochastic networks, it is further assumed that weights have a prior distribution $p_r\left(\Delta \mid \eta\right)$, where Δ denotes all the parameters in the network and η are prior hyper-parameters. The posterior distribution of neural network parameters conditioned on a set of IID examples $\{(x_i, y_i)\}_{i=1}^n$ with $X_n = \{x_i\}_{i=1}^n$ and $Y_n = \{y_i\}_{i=1}^n$ is typically given by an analytically intractable integral, with parameter-specific posterior probabilities $p\left(\Delta \mid X_n, Y_n\right)$ satisfying

$$\log p\left(\Delta \mid X_n, Y_n\right) \propto \log p_r\left(\Delta \mid \eta\right) + \sum_{i=1}^{n} \log p\left(y_i \mid x_i, \Delta\right) \ .$$

Variational inference [15–17, 37] is a technique for the approximation of posterior distributions involving analytically intractable integrals. It works by introducing a family of variational probability density functions $q\left(\Delta \mid \mu, \sigma\right)$, with μ and σ denoting variational parameters, such that a set of these specifies a family of probability distributions. Typically, the variational family is parametrically much simpler than the posterior distribution over network parameters $p\left(\Delta \mid X_n, Y_n\right)$. The main idea is to approximate the posterior $p\left(\Delta \mid X_n, Y_n\right)$ by optimizing a lower bound on the log-marginal likelihood of the model over the parameters of the variational distribution

$$\min_{q \in \mathcal{Q}} \ \mathrm{KL}\left(q \mid\mid p_r\right) - \sum_{i=1}^{n} \mathbb{E}_{\Delta \sim q(\Delta \mid \mu, \sigma)}\left[\log p\left(y_i \mid x_i, \Delta\right)\right], \quad (4)$$

where $\mathcal{Q}$ is a family of variational distributions specified by domains of parameters μ and σ. The Gaussian mean field approximation assumes that the variational distribution is the product of univariate Gaussian distributions, i.e. $q\left(\Delta \mid \mu, \sigma\right) = \prod_{i=1}^{p} \mathcal{N}\left(\Delta_i \mid \mu_i, \sigma_i^2\right)$, where p is the total number of parameters in the model, Δ_i is the i-th component of the parameter vector Δ, and $\mathcal{N}\left(\Delta_i \mid \mu_i, \sigma_i^2\right)$ is a univariate Gaussian distribution of Δ_i with mean μ_i and variance σ_i^2.

The expected log-likelihood of the model

$$L_n\left(q\right) = \sum_{i=1}^{n} \mathbb{E}_{\Delta \sim q(\Delta \mid \mu, \sigma)}\left[\log p\left(y_i \mid x_i, \Delta\right)\right]$$

is analytically intractable and an evaluation of this expectation is required for the forward-pass when computing the loss function for a setting of the variational parameters μ and σ. Stochastic variational inference approximates this term in the forward-pass by sampling the variational distribution [37]:

$$L_n\left(q\right) \approx \tilde{L}_m\left(q\right) = \frac{n}{m} \sum_{i=1}^{m} \log p\left(y_i \mid x_i, \Delta\right),$$

with $\Delta_j = \mu_j + \epsilon_j \sigma_j$ being a sample from $\mathcal{N}\left(\Delta_j \mid \mu_j, \sigma_j^2\right)$ given by $\epsilon_j \sim \mathcal{N}\left(\epsilon_j \mid 0, 1\right)$ $(1 \leq j \leq p)$, and where $\{(x_i, y_i)\}_{i=1}^m$ is a mini-batch with m random examples. As illustrated in Fig. 1 (the rightmost panel), the parameters of the neural network are populated with a random sample Δ drawn from the variational distribution and with that setting one computes the loss function for a particular mini-batch. The forward-pass sequence of actions is differentiable with respect to the variational parameters $\upsilon = \{(\mu_i, \sigma_i)\}_{i=1}^p$ and unbiased. Consequently, the gradient of this estimator is also unbiased and can be computed in the backward-pass by $\nabla_\upsilon L_n\left(q\right) \approx {}^{n}\!/_{m} \sum_{i=1}^{m} \nabla_\upsilon \log p\left(y_i \mid x_i, \Delta\right)$, where the network parameters Δ originate from the forward-pass components and are given by $\Delta_j = \mu_j + \epsilon_j \sigma_j$. Thus, stochastic neural networks update the variational mean and variance parameters during gradient descent and use back-propagation for the computation of the gradients with respect to these parameters. At test time, the parameters of neural architecture are populated with variational means. In this way, a stochastic neural network injects randomness into network parameters for each mini-batch. As a result, the inferred model can capture parameter uncertainty and is likely to be more stable to parameter perturbations than an equivalent deterministic model. A further regularization effect can be achieved via the Kullback–Leibler divergence term (Eq. 4), discussed in the next section.

A. Approximation of Kullback–Leibler Divergence

The Kullback–Leibler divergence term is responsible for regularization (Eq. 4) and it is defined in terms of an analytically intractable integral that is typically approximated by Monte Carlo estimates using samples from the variational distribution [15] or prior specific second order approximations [37, 38]. We propose an approximation scheme based on the Gauss–Hermite quadrature, which independently of the prior distribution used allows for an approximation with a polynomial of arbitrarily high degree. More specifically, variational inference typically relies on Gaussian mean field approximations and this implies that the divergence term can be expressed as a sum of one dimensional integrals with respect to univariate Gaussian measures. Such integrals can be effectively approximated using the Gauss–Hermite quadrature [39], which is a quadrature with the weight function $\exp(-u^2)$ over the interval $u \in (-\infty, \infty)$. The following theorem provides a formal specification of the Gauss–Hermite quadrature for univariate functions.

Theorem 2. *[Abramowitz and Stegun, 39] For a univariate function h and an integral*

$$\mathcal{J} = \int_{-\infty}^{\infty} h\left(u\right) \exp\left(-u^2\right) \, \mathrm{d}u \ ,$$

the Gauss-Hermite approximation of order s satisfies $\mathcal{J} \approx \sum_{i=1}^{s} w_i h\left(u_i\right)$, where $\{u_i\}_{i=1}^{s}$ are the roots of the physicist's version of the Hermite polynomial $H_s\left(u\right) = \left(-1\right)^s \exp\left(u^2\right) \frac{\mathrm{d}^s}{\mathrm{d}u^s} \exp\left(-u^2\right)$ and the corresponding weights $\{w_i\}_{i=1}^{s}$ are given by $w_i = \frac{2^{s-1} s! \sqrt{\pi}}{s^2 H_{s-1}(u_i)^2}$.

Such approximations have been studied theoretically, with convergence rates provided for polynomials and functions of limited regularity. More specifically, the Gauss–Hermite approximation of order s is exact and, thus, optimal for all polynomials of degree $2s-1$ or less [39]. For functions $h \in \mathcal{C}^{2s}$, the error of the Gauss–Hermite quadrature is given by [40]

$$\begin{aligned} &\mathcal{E}_s\left(h\right) = \\ &\int_{-\infty}^{\infty} h\left(u\right) \exp\left(-u^2\right) \, \mathrm{d}u - \sum_{i=1}^{s} w_i h\left(u_i\right) = \\ &\frac{s! \cdot \sqrt{\pi}}{2^s \cdot (2s)!} h^{(2s)}\left(\hat{u}\right) \ , \end{aligned} \tag{5}$$

where $\hat{u} \in (-\infty, \infty)$. Xiang and Bornemann [41] have studied convergence rates of the Gaussian quadrature for functions of limited regularity. The regularity of an integrand is expressed via the decay rate of its expansion coefficients in the basis formed by the Chebyshev polynomials of the first kind. In particular, if the expansion coefficients $a_i \in \mathcal{O}(i^{-p-1})$ for some $p > 0$ (where a_i corresponds to the Chebyshev polynomial of the i-th degree) then the error of the quadrature approximation of order s can be upper bounded by $\mathcal{O}(s^{-p-1})$ for $p > 2$. For $0 < p < 2$, on the other hand, the guaranteed convergence rate is sightly slower and can be upper bounded by $\mathcal{O}(s^{-3p/2})$. These results can provide theoretically well founded guidelines for selecting the approximation order and quantify the trade-offs between approximation quality and computational costs.

B. Illustration of the Approximation Scheme with Two Priors

In [37], it has been argued that *log-scale uniform* priors provide a theoretical justification for the dropout regularization technique [42] frequently used in the training of neural networks. The Bayesian aspect of that justification has recently been disputed in [43] but the technique can still be viewed as performing penalized log-likelihood estimation with the Kullback–Leibler divergence term acting as regularizer. The prior is given by $p_{r,\mathrm{lsu}}\left(\log\left|\Delta_i\right|\right) \propto \mathrm{const}$, or equivalently $p_{r,\mathrm{lsu}}\left(\left|\Delta_i\right|\right) \propto 1/|\Delta_i|$, where Δ_i is some network parameter. Two different second order approximations of the Kullback–Leibler divergence between Gaussian mean field posteriors and this prior distribution were provided in [37] and [38]. We propose an alternative Gauss–Hermite approximation, formalized in the following proposition. Just as in [42] and [37], we employ a parametrization of variational Gaussian mean field known as the *dropout posterior*, with mean parameter μ_j and variance $\sigma_j^2 = \alpha_j \mu_j^2$ specified via a scaling parameter $\alpha_j > 0$ (for all $1 \leq j \leq p$).

Proposition 3. *The* KL *divergence between a Gaussian distribution with the dropout parametrization of variance and a log-scale uniform prior can be approximated by*

$$\mathrm{KL}\left(q \,||\, p_{r,\mathrm{lsu}}\right) \approx -1/2 \log \alpha + 1/\sqrt{\pi} \sum_{i=1}^{s} w_i \log\left|v_i\right| + \mathrm{const.} \ ,$$

where $v_i = \sqrt{2\alpha} u_i + 1$ (for all $1 \leq i \leq s$) and the $\{u_i\}_{i=1}^{s}$ are roots of the Hermite polynomial with corresponding quadrature weights $\{w_i\}_{i=1}^{s}$.

Proof. From [37, Appendix C], we know that the Kullback–Leibler divergence term is given by

$$\mathrm{KL}\left(q \,||\, p_{r,\mathrm{lsu}}\right) = \mathbb{E}_{\mathcal{N}(\epsilon|1,\alpha)}\Big[\log\left|\epsilon\right|\Big] - \frac{1}{2}\log\alpha + \mathrm{const.}$$

The expectation with respect to the Gaussian random variable ϵ can be re-written as

$$\begin{aligned} &\mathbb{E}_{\mathcal{N}(\epsilon|1,\alpha)}\Big[\log\left|\epsilon\right|\Big] = \\ &\frac{1}{\sqrt{2\pi\alpha}} \int \exp\left(-\frac{\left(\epsilon-1\right)^2}{2\alpha}\right) \log\left|\epsilon\right| \, \mathrm{d}\epsilon = \\ &\frac{1}{\sqrt{\pi}} \int \log\left|\sqrt{2\alpha}t + 1\right| \exp\left(-t^2\right) \, \mathrm{d}t \ . \end{aligned}$$

The result now follows from Theorem 2 by taking $h\left(t\right) = \log\left|\sqrt{2\alpha}t + 1\right|$. □

The scale-mixture is another prior distribution frequently used in variational inference, first proposed in [15]. It resembles the so called spike and slab prior [44–46] and is given by

$$\begin{aligned} &p_{r,\mathrm{sm}}\left(\Delta_i \mid \xi, \eta_1, \eta_2, \lambda\right) = \\ &\lambda \cdot \mathcal{N}\left(\Delta_i \mid \xi, \eta_1^2\right) + \left(1-\lambda\right) \cdot \mathcal{N}\left(\Delta_i \mid \xi, \eta_2^2\right) \ , \end{aligned}$$

where Δ_i is a parameter of the model (see Eq. 4), η_1^2 and η_2^2 are prior (variance) hyper-parameters with $\eta_1 \ll \eta_2$, ξ is the prior mean, and $0 \leq \lambda \leq 1$ is the mixture scale. The hyper-parameters of the prior distributions (i.e., η_1, η_2, λ, and ξ) are kept fixed during optimization and can be chosen via cross-validation. The first mixture component is chosen such that $\eta_1 \ll 1$, which forces many of the variational parameters to concentrate tightly around the prior mean ξ (e.g., around zero for $\xi = 0$). The second mixture component has higher variance and heavier tails allowing parameters to move further away from the mean. The prior variance hyper-parameters are shared between all the network parameters and this is an important difference compared to approaches based on the spike and slab prior [46, 45, 44], where each model parameter has a different prior variance. The following proposition provides means for approximating the divergence term between a Gaussian mean field variational distribution and this prior function.

Proposition 4. *The* KL *divergence between a Gaussian distribution with the dropout parametrization of variance and a scale-mixture prior can be approximated by*

$$\begin{aligned} &\mathrm{KL}\left(q \,||\, p_{r,\mathrm{sm}}\right) \approx \\ &-\log\sqrt{2\pi\alpha\mu^2} - 1/\sqrt{\pi} \sum_{i=1}^{s} w_i \log p_{r,\mathrm{sm}}\left(v_i\right) - 1/2 \ , \end{aligned}$$

where $v_i = \left(\sqrt{2\alpha}u_i + 1\right)\mu$ *and the* $\{u_i\}_{i=1}^s$ *are roots of the Hermite polynomial with corresponding quadrature weights* $\{w_i\}_{i=1}^s$, α *and* μ *are variational parameters, and* $p_{r,\mathrm{sm}}$ *is some scale-mixture prior distribution.*

Proof. We can re-write the divergence term as

$$\begin{aligned}&\mathrm{KL}\left(q \,||\, p_{r,\mathrm{sm}}\right) = \\ &\int q\left(u\right)\log q\left(u\right)\,\mathrm{d}u - \int q\left(u\right)\log p_{r,\mathrm{sm}}\left(u\right)\,\mathrm{d}u = \\ &-H(q) - \mathbb{E}_q\left[\log p_{r,\mathrm{sm}}\left(u\right)\right],\end{aligned}$$

where $H\left(q\right)$ denotes the entropy of the univariate Gaussian distribution given by

$$q\left(u\right) = \frac{1}{\sqrt{2\pi\alpha\mu^2}}\exp\left(-\frac{\left(u-\mu\right)^2}{2\alpha\mu^2}\right)\ .$$

As the entropy of a Gaussian distribution defines an analytically tractable integral [e.g., see 47, 48], we have that the entropy of q is given by

$$H\left(q\right) = \log\sqrt{2\pi\alpha\mu^2} + 1/2\ .$$

On the other hand, the expected log-likelihood of the scale-mixture prior can be approximated using the Gauss-Hermite quadrature by observing that

$$\begin{aligned}&\mathbb{E}_q\left[\log p_{r,\mathrm{sm}}\left(u\right)\right] = \\ &\frac{1}{\sqrt{2\pi\alpha\mu^2}}\int\exp\left(-\frac{\left(u-\mu\right)^2}{2\alpha\mu^2}\right)\log p_{r,\mathrm{sm}}\left(u\right)\,\mathrm{d}u = \\ &\frac{1}{\sqrt{\pi}}\int\log p_{r,\mathrm{sm}}\left(\sqrt{2\alpha\mu^2}t + \mu\right)\exp\left(-t^2\right)\,\mathrm{d}t\ .\end{aligned}$$

The result now follows from Theorem 2 by taking $h\left(t\right) = \log p_{r,\mathrm{sm}}\left(\sqrt{2\alpha\mu^2}t + \mu\right)$. □

IV. Related Work

An alternative to learning a discriminative model with non-adaptive features is to learn these features automatically as part of a neural architecture that takes raw speech as input. In addition to having a more flexible inductive bias such a model would be less susceptible to the information loss that is inherent to waveform compression by means of a projection to a lower dimensional feature space [9, 49]. In particular, a model operating directly in the waveform domain has the potential to exploit local correlations within the signal that are typically discarded when computing Mel-filter bank values [50], as well as the information contained in a sequence of waveform samples without interruptions by frame boundaries characteristic to spectrograms and non-adaptive feature extraction techniques based on frame-based discrete Fourier transforms [51]. As a result of the latter, phonetic events on the boundaries of short frames are typically poorly described by filterbank features.

Whilst speech production embeds redundancies relevant for robustness, there are several challenges when dealing with these highly correlated raw speech inputs. In particular, the high dimensionality of waveform signals typically requires a larger number of parameters compared to standard features and a prolonged training time. Another difficulty is the fact that raw speech is known to be characterized by a large number of variations such as temporal distortion and speaker variability [11, 24]. Acoustic models based on neural networks operating directly in the waveform domain are, thus, likely to over-fit on small and moderately sized datasets without appropriate inductive bias. In this sense our approach, which combines variational inference with Lipschitz continuity of the operator mapping, provides a theoretical underpinning for the design and learning of effective waveform-based acoustic models. Previous work has also resorted to similar techniques for maintaining the balance between dataset size and model complexity. Watanabe et al. [19, 52] have used variational inference for clustering of states in triphone hidden Markov models (HMM) and learning the appropriate number of components in Gaussian mixture models (GMM). In contrast to this, we use variational inference to learn a stochastic convolutional network that models the conditional probability of a triphone state-id given an input waveform frame.

Graves [17] and Braun and Liu [20] have used variational inference to learn a recurrent neural network as part of an end-to-end acoustic model. While the latter approach does not have an explicit KL divergence term characteristic to variational inference, there is a sparsity inducing penalty over the parameters defining standard deviations, which under a suitable prior could be seen as an instance of KL divergence. In both of these works it was observed that parameter uncertainty is correlated with the importance of individual parameters for the speech recognition tasks considered. Similarly, Hu et al. [53] have proposed a Bayesian neural network that allows for learning with more expressive activation functions in the context of multi-layer perceptrons and standard recurrent neural networks. In particular, each hidden layer of the model relies on Bayesian averaging relative to a weight prior when computing the corresponding outputs, and variational inference for dealing with the resulting analytically intractable integrals. A Bayesian approach coupled with variational inference has also been used in [54] for speaker adaptation. The main difference to this line of work is that neither of those models operates in the waveform domain, but rely on low-dimensional feature spaces generated by FBANK or MFCC features. This allows for scalable inference of recurrent models, which is known to be computationally expensive for high dimensional inputs such as waveform signals. Moreover, prior work in speech recognition (to the best of our knowledge) considers variational inference independently of Lipschitz continuity and other design principles that could allow for learning of robust models in small scale settings. Recently, an approach for modulation filter-learning based on an encoder-decoder architecture and variational inference has been considered in [55] and [56]. The encoder takes as input a Mel-spectrogram constructed using speech segments of fixed length and learns its latent representation. The optimization of encoder-decoder parameters is performed using variational inference and the learned filters are then used to generate features that are used as input to an MLP. In contrast to this, we use variational inference to learn filters jointly with other network parameters (i.e., filterbank-based feature extraction/learning is not done independently of training other network modules).

A common characteristic of previous approaches for waveform-based speech recognition is the use of relatively large datasets [11, 12]. In such a regime, waveform-based acoustic models are competitive with architectures relying on standard features (i.e., MFCC, FBANK, and FMLLR). Another difference compared to our approach is that previous architectures typically employ a convolutional layer with weighted ℓ_1 or ℓ_2 pooling (25 ms long frames) to emulate filterbank features and reduce the dimension of the representation quickly [50, 57]. In contrast to this, we perform gradual compression of the waveform sub-band decomposition via max pooling and thus overcome the information loss inherent in standard features. Moreover, we use the RELU non-linearity throughout the network and do not apply the LOG operator to the outputs of the initial block. Sainath et al. [11] propose an architecture that takes raw speech inputs and applies one-dimensional convolutions first in the time-domain and then the frequency-domain, designed to extract band-pass features from the waveform. The architecture itself is a recurrent net that requires more than $2,000$ hours of training data to match the performance of models with standard features. Similarly, Zhu et al. [12] combine two convolutional layers with recurrent blocks in end-to-end training, requiring more than $2,400$ hours of training data for state-of-the-art results. Ghahremani et al. [24] proposed a feedforward architecture based on a convolutional feature extraction layer, with the outputs of that block passed to a deep time-delay neural network (TDNN). The empirical results indicate that the approach is competitive with MFCC-based architectures on large datasets. It has not been evaluated on noisy speech and it is unclear how well it would generalize from small datasets.

Our architecture performs parametric sub-band decomposition of speech waveforms and it is most closely related to SINCNET [23], which employs three 1D convolutional layers on top of the parametric block. SINCNET is considered to be the state-of-the art model for waveform-based speech recognition. A related architecture is SINC^{2}NET; this links a parametric convolution block to an MLP [58]. Recently, complex-valued parametric filters have been used to initialize a complex non-parametric convolution block in a deep network for end-to-end speech recognition [59–61]. In comparison to [59], we show that our approach generalizes better on the small TIMIT dataset. In our experiments, we use the SINCNET architecture (code available) as a representative baselines from this class.

Recently, an approach based on concatenation of multiple convolutional blocks was proposed [22], in which convolutional blocks capture different contexts in time and learn band-pass filters that are more expressive than classic Mel-filterbanks, which operate on a single fixed context. The approach was evaluated on both noisy and conversational speech. In our experiments, we compare to this baseline and demonstrate statistically significant improvement on the AMI-IHM dataset (12% relative).

V. EXPERIMENTS

We evaluate the proposed approach with a series of experiments on three different datasets: TIMIT [62], AURORA4 [63], and AMI-IHM [64]. In all the experiments[1], we train a context dependent hybrid HMM model based on frame labels (i.e., HMM state ids) generated using a triphone model from Kaldi [65] with 25 ms frames and 10 ms stride between the successive frames. The data splits (train/validation/test) originate from the Kaldi framework. In the pre-processing step, we assign the Kaldi frame label to the 200 ms long segment of raw speech centered at an original Kaldi frame (keeping 10 ms stride between the successive frames of raw speech). To be consistent with our baselines on TIMIT, we generate frame labels using the DNN triphone model and decoding configuration from [23]. For AURORA4, on the other hand, we generate frame labels using both GMM and DNN triphone models, relying on the default decoder configuration from Kaldi.

We describe below four sets of experiments. The first aims at demonstrating the impact of particular design choices on the effectiveness of acoustic models. More specifically, our empirical results show that: modulation filter learning can improve the performance of acoustic models in a statistically significant way (subsection A, below), the proposed approximation scheme for the Kullback–Leibler divergence term is generally more effective than previous approaches (subsection B, below), modulation filter learning moves away from the initial solution and converges to different distributions of modulation frequencies for different learning tasks (subsection C, below), and probabilistic parametrization of the neural architecture contributes to a 7.4% relative improvement in the error rate compared to the deterministic one (subsection D, below). The second set of experiments (subsections D and E, below) is aimed at showing that the proposed approach does not over-fit on what is considered to be a small dataset in speech recognition (i.e. TIMIT). Moreover, the results also indicate that a combination of variational inference and Lipschitz continuous architectures for waveform-based speech recognition such as PARZNETS does not require large training datasets to outperform models based on standard filterbank features. The third experiment (subsection E, below) deals with noisy speech and shows that the proposed approach can learn an effective noise robust representation of waveform signals. The fourth and final experiment aims at demonstrating the effectiveness of the proposed approach on conversational speech (i.e., AMI-IHM), with approximately 80 hours of audio. The experiment shows a clear improvement over recently proposed waveform-based approaches (12% relative) and a competitive performance relative to filterbank architectures known for their effectiveness on this dataset. We also observe that variational inference consistently contributes to an improvement in the error rate compared to the deterministic models.

A. Can modulation filter learning improve the effectiveness of waveform-based acoustic models?

The goal of this experiment is to demonstrate that filter optimization can be more effective than non-adaptive filtering of speech signals, in a way that is statistically significant. To that end, we train two neural networks with identical architectures

[1]A detailed setup of our experiments along with the source code can be found in the project repository https://bitbucket.org/doglic/asr/.

TABLE I: The table reports the average phoneme error rates (standard deviations are provided in the brackets), obtained using variational PARZNETS 1D and Gaussian mean field (variational) inference on the TIMIT dataset.

SAMPLE	VI – LOG-SCALE UNIFORM				VI – SCALE MIXTURE	
	SQUARED EPANECHNIKOV			GAUSS	SQUARED EPANECHNIKOV	
	NON-ADAPTIVE MEL-FILTERS KL APPROXIMATION: HERMITE-GAUSS QUAD.	ADAPTIVE FILTERS KL APPROXIMATION: HERMITE-GAUSS QUAD.	ADAPTIVE FILTERS KL APPROXIMATION: MOLCHANOV ET AL [38]	ADAPTIVE FILTERS KL APPROXIMATION: HERMITE-GAUSS QUAD.	ADAPTIVE FILTERS KL APPROXIMATION: HERMITE-GAUSS QUAD.	ADAPTIVE FILTERS KL APPROXIMATION: MCMC [15]
DEV	15.02 (±0.26)	14.95 (±0.14)	**14.77** (±0.15)	14.83 (±0.13)	15.64 (±0.11)	15.58 (±0.20)
TEST	16.95 (±0.25)	**16.52** (±0.22)	16.63 (±0.23)	16.60 (±0.22)	17.41 (±0.17)	17.56 (±0.16)

TABLE II: AURORA4, word error rates obtained using different test samples.

	VI – LOG-SCALE UNIFORM				VI – SCALE MIXTURE	
	8 X CNN			10 X CNN	8 X CNN	
ADAPTIVE FILTERS		✓	✓	✓	✓	✓
KL: HERMITE-GAUSS	✓	✓		✓	✓	
KL: MOLCHANOV ET AL.			✓			
KL: MCMC						✓
A. SAME MICROPHONE						
CLEAN (A)	3.05	2.88	2.84	2.78	3.12	**2.71**
B. SAME MICROPHONE						
CAR	3.29	3.34	3.14	**3.10**	3.29	3.25
BABBLE	4.63	4.33	4.84	**4.26**	4.54	4.84
RESTAURANT	6.46	**6.00**	6.18	6.54	6.65	6.37
STREET	5.87	5.87	5.88	**5.70**	6.22	6.16
AIRPORT	4.76	4.45	4.58	**4.43**	4.78	4.61
TRAIN	6.41	6.33	**6.30**	6.35	**6.30**	6.35
AVERAGE (B)	5.24	**5.05**	5.15	5.06	5.30	5.26
C. DIFFERENT MICROPHONES						
CLEAN (C)	5.90	5.59	6.02	**5.27**	6.09	5.96
D. DIFFERENT MICROPHONES						
CAR	9.79	9.30	9.36	**9.10**	9.84	10.14
BABBLE	15.84	15.41	16.01	**14.78**	16.07	16.16
RESTAURANT	20.08	20.77	21.39	**19.56**	21.15	21.24
STREET	17.31	**16.80**	17.71	17.28	17.65	18.61
AIRPORT	14.70	13.88	14.65	**13.30**	14.70	14.94
TRAIN	17.43	**16.99**	17.49	17.07	17.64	17.90
AVERAGE (D)	15.86	15.53	16.10	**15.18**	16.18	16.50
AVERAGE (ALL)	9.68	9.42	9.74	**9.25**	9.86	9.95

(see Fig. 1) using variational inference with the Kullback–Leibler divergence term approximated via the Hermite–Gauss (HG) quadrature: *i*) a neural network with non-adaptive Parzen filters initialized just as in Mel-frequency coefficients (denoted with MEL-FILTERS in Tables I and II), and *ii*) the joint filter and neural network learning proposed in this work (see ADAPTIVE FILTERS, HERMITE-GAUSS QUAD. under log-scale uniform prior VI in Tables I and II). The Parzen filters of the latter adaptive operator are initialized exactly as the non-adaptive ones. To assess whether one method performs statistically significantly better than the other on TIMIT, we perform the paired Welch t-test [66] based on 5 repetitions of the experiment. The t-test indicates that filter learning is with 90% confidence statistically significantly better than non-adaptive filtering. We similarly studied performance on AURORA4, which is a much larger dataset than TIMIT where repeated training is time consuming and expensive. However, the dataset contains 14 different test samples and this allows us to employ the Wilcoxon signed rank test [67, 68] to again establish whether one approach is statistically significantly better than the other. The test indicates that filter learning is with 95% confidence statistically significantly better than non-adaptive filtering on AURORA4 (see e.g. Table II).

B. How effective is the Gauss–Hermite approximation scheme?

Having established that modulation filter learning can be significantly better than static filtering, we proceed to show that Hermite–Gauss quadrature is an effective scheme for the approximation of the Kullback–Leibler divergence term acting as a regularizer in variational inference. In particular, we compare the effectiveness of neural networks learned via variational inference and existing strategies for approximation of the Kullback–Leibler divergence term, defined using the log-scale uniform [38] and scale mixture priors [15]. Table I (see SQUARED EPANECHNIKOV modulation filters, TEST sample) provides the results on TIMIT and shows that the approximation based on the Hermite–Gauss quadrature (see HERMITE-GAUSS QUAD. columns) is on average better than existing approximation schemes (see MOLCHANOV ET AL. and MCMC columns). However, the Welch t-test does not show a statistically significant improvement of the Hermite–Gauss quadrature over the alternatives on this dataset. Table II summarizes our results on AURORA4 and demonstrates a significant improvement over the baselines when using the Hermite–Gauss quadrature to approximate the Kullback–Leibler divergence term. More specifically, the Wilcoxon signed rank test in the case of log-scale uniform prior shows that the approximation based on the Hermite–Gauss quadrature is with 95% confidence statistically significantly better than the state-of-the-art approximation proposed in [38].

C. Do modulation frequencies move away from the initial solution and converge to different distributions for different learning tasks?

The goal of this experiment is to demonstrate that the optimization of modulation filters changes the initial distribution of modulation frequencies and bandwidths. Fig. 2 provides a comparison of kernel density estimators for modulation frequencies and filter bandwidths. From the figure, it is evident that the initial and optimized distributions are quite different for filter bandwidths on both datasets. Moreover, there is an interesting difference between the distributions of modulation frequencies between TIMIT and AURORA4 datasets, which might be due to multi-condition training and various noise conditions characteristic to AURORA4.

D. How does the approach fare relative to state-of-the-art feedforward models on TIMIT*?*

Table III summarizes our empirical results in comparison to state-of-the-art feedforward architectures on TIMIT. In addition to the lowest obtained error rate (denoted with MIN), we also report the average result over 5 simulations. A comparison to previously reported results for waveform-based speech recognition indicates that our approach performs the best on

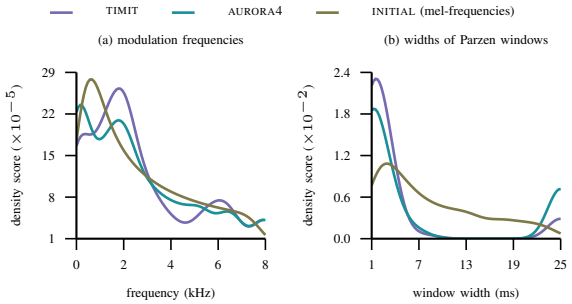

Fig. 2: Comparison of the initial distributions of modulation frequencies and bandwidths to those at the end of the training process.

TABLE III: Comparison of phoneme error rates obtained in our experiments on TIMIT to the ones reported for relevant feedforward nets.

METHOD	AVG	MIN
A. RAW SPEECH BASELINES (OPTIMIZED FILTERS)		
VARIATIONAL PARZNETS	**16.5**	**16.2**
DETERMINISTIC PARZNETS	17.7	17.5
SINCNET [23, 69]	17.5	17.2
SINC^{2}NET [58]	–	16.9
END-TO-END CNN [59]	–	18.0
RAW SPEECH CNN [23]	18.3	18.1
B. STANDARD FEATURES (NON-ADAPTIVE FILTERS)		
FMLLR + MLP	16.9	16.7
MFCC + MLP [70]	18.1	17.8
MULTI-RES DSS + CNN & MLP [71]	–	17.4

average on this task. Moreover, this is the first such approach that outperforms all the feedforward architectures built on top of standard non-adaptive features. Our results also show that variational inference contributes to a 7.4% relative improvement on this dataset over a deterministic network with identical architecture (see DETERMINISTIC PARZNETS in Table III). We note here that recent work has reported lower error rates on TIMIT using recurrent nets and statically extracted features. In particular, [72] reports the following error rates for gated recurrent units (GRU): LI-GRU 15.8% and LI-GRU FMLLR 14.8%. In the waveform domain with low-resources (i.e., small datasets such as TIMIT) recurrent nets perform worse than feedforward models. In particular, our best result on this dataset with recurrent nets in the waveform domain was 18.8%, which is significantly worse than the best observed result with PARZNETS (i.e., 16.2%). The good performance of models based on FMLLR features should not come as a surprise, because that feature extraction technique performs speaker and domain adaptation as well. Our future work will explore recurrent architectures in the waveform-domain, combined with regularization mechanisms provided by variational inference.

E. How does the approach fare relative to state-of-the-art feedforward models on AURORA4*?*

AURORA4 is a medium vocabulary task based on clean speech from the Wall Street Journal (WSJ0) corpus [73]. The clean speech was corrupted by six different noise types at different SNRs. The test sets consist of noise corrupted utterances recorded by a primary and a secondary microphone. In Table IV we provide a summary of our results on this dataset relative to state-of-the-art feedforward architectures.

TABLE IV: Word error rates obtained on AURORA4 using multi-condition training and input/context frames of 200 ms (A: clean speech with same microphone, B: noisy speech with same microphone, C: clean speech with different microphones, D: noisy speech with different microphones).

METHOD	A	B	C	D	AVG
A. RAW SPEECH & VAR. BASELINES (OPTIMIZED FILTERS)					
DNN ALIGNMENTS					
VAR. PARZNETS (10 X CNN1D)	2.22	4.50	4.71	14.72	**8.73**
DET. PARZNETS (10 X CNN1D)	2.35	4.73	4.86	15.48	9.17
VAR. PARZNETS (8 X CNN1D)	2.15	4.50	5.28	15.07	8.92
DET. PARZNETS (8 X CNN1D)	2.24	4.61	5.75	15.48	9.18
GMM ALIGNMENTS					
VAR. PARZNETS (10 X CNN1D)	2.78	5.06	5.27	15.18	9.25
VAR. PARZNETS (8 X CNN1D)	2.88	5.05	5.59	15.53	9.42
SINCNET [69]	3.42	6.33	6.13	16.99	10.68
CVAE FEATS + MLP [55, 56]	3.50	7.40	6.90	17.10	11.20
B. STANDARD FEATURES (NON-ADAPTIVE FILTERS)					
FBANK + VD10 X CNN2D [21]	4.13	6.62	5.92	14.53	9.78
FBANK + VD8 X CNN2D [21]	3.72	6.57	5.83	14.79	9.84
FMLLR + MLP	3.34	6.27	5.74	16.04	10.21
MFCC + MLP	4.28	7.44	8.73	18.71	12.14
DSS (UTT. NORM.) + JUNCT. NET	3.05	5.82	6.11	15.94	9.98
DSS (W/O NORM.) + JUNCT. NET	4.09	6.35	8.24	19.07	11.78

The first experiment compares our approach (8 X CNN1D) to the state-of-the-art architecture for waveform-based speech recognition [23, SINCNET] and shows a statistically significant [68, 67, Wilcoxon test, 95% confidence] improvement over that baseline. We also compare to a recent approach for modulation filter-learning using encoder-decoder architecture and variational inference [55, 56]. The results again show (with 95% confidence) that the proposed approach is statistically significantly better than the baseline from [55, 56]. Following this, we compare our results to the error rates reported in [21] for 8 and 10-layer deep 2D convolutional networks (VDCNN2D) based on statically extracted features using 200 ms long raw-speech segments (i.e., 17 FBANK frames). This might be an unfair comparison to our approach, because we use the less expressive 1D convolutions in our architecture. Still, the results indicate that the variational PARZNETS architecture with 8 convolutional layers outperforms significantly the network with 10 CNN2D layers from [21]. Furthermore, we extend our architecture (Fig. 1) to 10 convolutional layers by employing time-padding in 1D convolutions to allow for another double convolutional block. The results indicate a further improvement in accuracy as a result of this modification.

Another particularly interesting observation is that the gains of our approach over noisy samples do not come as a result of performance degradation on clean speech. We note here that [21] reports a slightly better error rate with 2D convolutions and FBANK features when the context size is increased to 250 ms (i.e. 21 frames), in combination with time and frequency padding (WER 8.81%). Table IV (see DNN ALIGNMENTS) shows that our approach provides a competitive error rate (WER 8.73%) with smaller context size (i.e., 200 ms) and less expressive time-padded 1D convolutions. Moreover, a recent approach based on multi-octave convolutions and 15 such convolutional layers has achieved the error rate of 8.31% on this dataset [74].

In a follow up work [75], we have investigated PARZNETS with 2D convolutional operators coupled with Bernoulli dropout layers (i.e. a special case of stochastic neural networks with variance parameter fixed over an entire network layer). This

TABLE V: The word error rates obtained on dev and eval sets of AMI-IHM with various input features and neural architectures. We did not use any data augmentation techniques or *i-vectors* in the experiments. Following the original Kaldi recipe, a 3-GRAM language model built from the AMI and FISHER data was adopted. Some of the related baselines relied on a contextually more expressive 4-GRAM language model, and were compiled solely using the AMI data. The column SIZE refers to an approximate number of differentiable parameters in the respective neural architectures.

ARCHITECTURE	DEV	EVAL	LM	SIZE
A. RAW SPEECH BASELINES (ADAPTIVE FILTERS)				
VAR. PARZNETS (10 X CNN1D)	**24.7**	**25.7**	3-GRAM	17.4 M
DET. PARZNETS (10 X CNN1D)	25.0	26.4	3-GRAM	8.7 M
VAR. PARZNETS (8 X CNN1D)	25.1	26.4	3-GRAM	19.0 M
DET. PARZNETS (8 X CNN1D)	25.9	27.7	3-GRAM	9.5 M
SINCNET [77]	28.0	30.2	3-GRAM	9.0 M
MULTI-SPAN-DNN [22]	27.2	29.3	4-GRAM	4.7 M
B. STANDARD FEATURES (NON-ADAPTIVE FILTERS)				
FBANK-MLP [22]	28.3	31.1	4-GRAM	3.0 M
FMLLR-MLP	26.0	27.1	3-GRAM	8.5 M
TDNN [78]	25.3	26.0	3-GRAM	7.7 M

approach achieved a word error rate of 7.80%, which is the best reported number on this dataset for waveform-based speech recognition. Here, it is important to note that 1D PARZNETS baselines from [75] employ time-padded convolutions and an extra fully connected layer in the MLP block compared to the neural architecture considered in this paper.

In addition to waveform-based baselines and deep convolutional networks operating with standard non-adaptive features, we have also compared our approach to a junction network [71] coupled with first and second order deep scattering spectrum features (see Table IV, DSS + JUNC. NET). The latter is a non-adaptive wavelet-based feature extraction technique [28] that generates features of different orders, with the first order coefficients approximately equal to MFCC, and higher order coefficients recovering information lost at lower levels. Our experiments demonstrate that PARZNETS can outperform this approach, even when it is supplied with utterance level normalization. In parallel with this work, we have also proposed deep scattering power spectrum features [76]. The latter non-adaptive feature extraction technique coupled with the junction neural architecture and utterance level normalization performs on par with PARZNETS (WER 8.83%). Given that deep scattering spectrum recovers information lost at lower levels, we hypothesize that this might be yet another indication for the relevance of information loss (characteristic to standard filterbank features) for robustness to standard noise corruptions.

F. How does the approach fare relative to state-of-the-art raw waveform baselines on AMI-IHM*?*

AMI-IHM is a conversational speech dataset with approximately 80 hours of speech, recorded using individual headset microphones. The alignments were generated using the Kaldi recipe configured with $3,984$ HMM state ids. Table V summarizes our result relative to relevant baselines on this dataset.

We have first compared variational PARZNETS with 8 and 10 convolutional layers to two recently published raw waveform approaches for this task: multi-span raw waveform models [22] and SINCNET [77]. Our empirical results show that variational PARZNETS advance the state-of-the-art in waveform-based acoustic models on this dataset, with over 12% relative improvement in WER compared to these baselines. Moreover, we also compare to deep time-delay neural networks [TDNN, 78] based on FBANK features (considered to be the state-of-the-art feedforward model on this dataset) and show that variational inference coupled with a PARZNETS architecture (10 X CNN1D) can outperform that approach. We note here that we have not used any data augmentation or i-vectors in our experiments, both techniques which could be combined with our approach and are known to further improve the accuracy on this dataset.

Finally we note that our experiments were conducted using a cross entropy (CE) loss function. Experiments using a sequence discriminative approach (LF-MMI) indicate that the WERs could be further lowered – Povey et al [79] indicated that using LF-MMI in place of CE can reduce the error rate by about 10% relative, and more recently a regularised LF-MMI training with significant data augmentation (6x) resulted in a WER of 18.0% on this task [80].

VI. DISCUSSION

This section discusses some of the model choices and assumptions made by our approach. We also address the empirical evaluation and the ablation studies that we have performed to discern the effects of individual components of our approach.

The proposed approach employs a variational family of univariate Gaussian distributions, known as the mean field assumption. While such a variational family might be perceived as overly simplistic, recent work [81] has demonstrated that deep Bayesian/stochastic neural networks equipped with univariate Gaussian distributions can build complex covariance structures through multiple layers. The proposed neural architecture combines 8-10 convolutional layers with multi-layer perceptrons and, thus, provides sufficient depth.

The main reason for selecting the probabilistic formulation of the neural architecture is to enforce the bounded weight property across the network and, thus, allow for learning of a robust acoustic model with a good Lipschitz constant. Variational inference alone, however, is not necessary to guarantee bounded weights across the neural network. That property will depend on the choice of prior function and holds for the Gaussian and scale-mixture priors. For the log-scale uniform prior, Section III-A provides a brief discussion and reference to relevant related work where it has been demonstrated that learning with that prior amounts to performing penalized log-likelihood estimation, with the Kullback–Leibler divergence term responsible for regularization. Moreover, the dropout regularization technique [42] can be theoretically justified as variational inference with the log-scale uniform prior. Hence, the proposed approach exploits means to generalize the most frequently used regularization method for neural networks. Our experiments, however, demonstrate that Gaussian and scale-mixture priors do not provide a good inductive bias for waveform-based acoustic models. Future work will explore the potential of more complex prior functions.

In our ablation study (see Section V-A), we have compared the effectiveness of two identical architectures, one with modulation filter learning and the other with a priori fixed

or non-adaptive filters. Our empirical results indicate that filter learning can be statistically significantly more effective than non-adaptive filters. Moreover, Fig. 2 shows that modulation frequencies converge to different distributions for different learning tasks and this is yet another indication that non-adaptive filters do not provide a universally optimal inductive bias. When evaluating the effectiveness of the approach relative to standard features such as FBANK and MFCC one should bear in mind that different feature representations require different neural architectures and inductive biases for state-of-the-art results. Moreover, there is a significant difference in the dimension of the inputs to neural networks operating with raw waveforms on the one hand and FBANK or MFCC features on the other, because of the aggressive compression performed by the latter. In addition to this, neural networks operating with statically extracted features typically encode more information into the training process by means of speaker and utterance level normalizations, which are known to improve the performance of acoustic models. To make the comparison between different feature representations fair, we have decided to compare our approach to state-of-the-art feedforward architectures operating in low-dimensional feature spaces. Tables III and IV indicate a competitive performance of our approach relative to state-of-the-art baselines based on statically extracted features. Moreover, the approach is more effective than any other waveform-based approach and in this sense advances the state-of-the-art.

We conclude with a reference to the selected filterbank, which is simple to implement and provides the band-pass properties required to establish the Lipschitz continuity of the waveform-based operator mapping. The parametrization allows for an independent control over bandwidth and modulation frequency, which is sufficient to emulate a sub-band decomposition as in standard statically extracted features. In Table III (see RAW SPEECH CNN and END-TO-END CNN), we have compared to deep convolutional networks that employ modulation filter learning with a standard non-parametric convolutional layer. Our empirical results indicate that the strong inductive bias encoded via a parametric convolutional layer can lead to more effective acoustic models, especially in low-resource settings.

CONCLUSION

We have outlined a principled framework for learning effective waveform-based acoustic models. The framework combines stochastic variational inference with a Lipschitz continuous architecture/operator that learns to gradually extract relevant features. The approach operates directly in the waveform domain to avoid potential information loss inherent to standard feature extraction techniques such as MFCC and FBANK coefficients. In our experiments, the approach outperforms recently proposed architectures for waveform-based speech recognition (e.g., SINCNET) as well as a relevant deep convolutional networks for learning of robust acoustic models using FBANK features [21]. Moreover, our empirical results show that the proposed approach allows for learning of effective acoustic models using relatively small datasets. Our future work will explore the potential of stochastic recurrent architectures operating in the waveform domain as well as different priors that could further improve the inductive bias via the regularization mechanism provided by the Kullback–Leibler divergence term. To the best of our knowledge, this is the first time that a variational approach has achieved results competitive with state-of-the-art on continuous speech recognition.

APPENDIX A
TRAINING PROCEDURE

In all the experiments, the minibatch size was set to 256 samples. For our deterministically trained baselines, we tried two batch sizes, 256 and 128, and report the better of the two error rates in our tables. The feature extraction parameters involving Parzen filters and convolution layers that synthesize features across filtered signals were optimized using the RMSPROP algorithm [82] with initial learning rate 0.0008. The fully connected blocks were optimized using the standard stochastic gradient descent with initial learning rate 0.08. This combination of optimization algorithms (with all the blocks trained jointly) has been found to be the most effective, confirming the findings in [23]. Alternative algorithms that were tried and found to be too aggressive (providing lower training error but worse generalization) were ADAM [83], NADAM [84] and SGD with momentum. Here, it is important to note that the conclusions of our ablation studies were consistent under changes to the optimization algorithm. The learning rates were decreased by a factor of $1/2$ if at the end of an epoch the relative improvement in validation error was below a specified threshold (e.g., 0.1% for the frame classification error). Moreover, if the validation error degraded then training was continued using the model from the previous epoch (with learning rates again decreased by a factor $1/2$). We terminate the training process after at most 25 epochs or upon observing no improvement in the validation error for 3 successive epochs.

In previous work [85, 38] it was established that, for some priors, stochastic variational inference tends to trim too many parameters in the early stages of the training. To address this issue it was proposed [85] to rescale the Kullback–Leibler regularization term with a hyperparameter ρ_t such that $\rho_{t+1} = \min\{1, \rho_t + c\}$ with $\rho_0 = 0$ and some constant $0 < c < 1$ (e.g., $c = 0.2$), and where t denotes the epoch number (starting from $t = 0$). We followed this heuristic in all of our experiments and observed an improvement in accuracy. Following the findings in [86], we also considered two notions of validation error in our preliminary experiments (omitted here for brevity) classification error of raw-speech frames and entropy regularized log-loss [86]. The empirical results from [86] indicate that the latter error correlates better with the token error rate of continuous speech recognition. Indeed, our best results were obtained using the entropy regularized log-loss as the validation objective. Just as in [15], we observed an improvement in accuracy for models trained using batch-specific importance weighting of the divergence term. However, the cooling schedule proposed in [15, Eq. 9] was too strong for the datasets considered here because of the much larger number of batches. To address this, we replaced base 2 proposed in [15] with another constant, computed such that the minimal importance weight is equal to machine precision for 32-bit

floating point arithmetics. In addition to these findings we also observed that in some cases the optimization (overly) focuses on the maximization of the log-likelihood for the already correctly classified speech frames. To mitigate this and ensure that the optimization objective is always bounded, we transformed softmax probabilities (denoted with p) by

$$\log p \quad \rightarrow \quad \log\left((1-2\kappa)\,p+\kappa\right)\ , \tag{6}$$

with κ denoting a small jitter constant (e.g. $\kappa = 10^{-8}$).

ACKNOWLEDGMENTS

This work was supported in part by EPSRC grant EP/R012067/1. The authors would also like to thank Steve Renals and Peter Bell for valuable discussions and comments that have improved the manuscript. The Kaldi alignments were generated with the help of Erfan Loweimi and Neethu Joy.

REFERENCES

[1] F. Li, A. Trevino, A. Menon, and J. Allen, "A psychoacoustic method for studying the necessary and sufficient perceptual cues of american english fricative consonants in noise," *The Journal of the Acoustical Society of America*, vol. 132, 2012.

[2] C. Moore, T. Lee, and F. Theunissen, "Noise-invariant neurons in the avian auditory cortex: Hearing the song in noise," *PLOS Computational Biology*, vol. 9, no. 3, 2013.

[3] Z. Tüske, R. Schlüter, and H. Ney, "Acoustic modeling of speech waveform based on multi-resolution, neural network signal processing," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018.

[4] J. Bridle and M. Brown, "An experimental automatic word-recognition system," JSRU, Ruislip, UK, Tech. Rep. 1003, 1974.

[5] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1980.

[6] L. Alsteris and K. Paliwal, "Further intelligibility results from human listening tests using the short-time phase spectrum," *Speech Communication*, vol. 48, 2006.

[7] B. Meyer, M. Wächter, T. Brand, and B. Kollmeier, "Phoneme confusions in human and automatic speech recognition," *INTERSPEECH*, 2007.

[8] S. Peters, P. Stubley, and J.-M. Valin, "On the limits of speech recognition in noise," *IEEE International Conference on Acoustics, Speech and Signal Processing*, 1999.

[9] M. Ager, Z. Cvetkovic, and P. Sollich, "Combined waveform-cepstral representation for robust speech recognition," *IEEE International Symposium on Information Theory*, 2011.

[10] J. Yousafzai, P. Sollich, Z. Cvetkovic, and B. Yu, "Combined features and kernel design for noise robust phoneme classification using support vector machines," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, pp. 1396–1407, 2011.

[11] T. Sainath, R. Weiss, K. Wilson, A. Senior, and O. Vinyals, "Learning the speech front-end with raw waveform CLDNNs," in *INTERSPEECH*, 2015.

[12] Z. Zhu, J. Engel, and A. Hannun, "Learning multiscale features directly from waveforms," in *INTERSPEECH*, 2016.

[13] D. Barber, *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 2012.

[14] Z. Ghahramani, "Probabilistic machine learning and artificial intelligence," *Nature*, vol. 521, no. 7553, pp. 452–459, 2015, on Probabilistic models.

[15] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight uncertainty in neural network," in *Proceedings of the 32nd International Conference on Machine Learning*, 2015.

[16] W. Buntine and A. Weigend, "Bayesian back-propagation," *Complex Systems*, 1991.

[17] A. Graves, "Practical variational inference for neural networks," in *Advances in Neural Information Processing Systems 24*. Curran Associates, Inc., 2011.

[18] S. Watanabe and A. Nakamura, "Bayesian approaches to acoustic modeling: a review," *APSIPA Transactions on Signal and Information Processing*, vol. 1, 2012.

[19] S. Watanabe, Y. Minami, A. Nakamura, and N. Ueda, "Bayesian acoustic modeling for spontaneous speech recognition," 2003.

[20] S. Braun and S. Liu, "Parameter uncertainty for end-to-end speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019.

[21] Y. Qian, M. Bi, T. Tan, and K. Yu, "Very deep convolutional neural networks for noise robust speech recognition," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, 2016.

[22] P. von Platen, C. Zhang, and P. Woodland, "Multi-span acoustic modelling using raw waveform signals," in *INTERSPEECH*, 2019, pp. 1393–1397.

[23] M. Ravanelli and Y. Bengio, "Speech and speaker recognition from raw waveform with SincNet," *arXiv:1812.05920*, 2018.

[24] P. Ghahremani, V. Manohar, D. Povey, and S. Khudanpur, "Acoustic modelling from the signal domain using CNNs," in *INTERSPEECH*, 2016.

[25] S. Mallat, "Understanding deep convolutional networks," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 2016.

[26] F. Croce and M. Hein, "Provable robustness against all adversarial l_p-perturbations for $p \geq 1$," in *International Conference on Learning Representations*, 2020.

[27] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[28] J. Andén and S. Mallat, "Deep scattering spectrum," *IEEE Transactions on Signal Processing*, 2014.

[29] S. Mallat, "Group invariant scattering," *Communications on Pure and Applied Mathematics*, 2012.

[30] M. Gales and S. Young, "The application of hidden Markov models in speech recognition," *Foundations and Trends in Signal Processing*, 2007.

[31] E. Parzen, "On estimation of a probability density function and mode," *The Annals of Mathematical Statistics*, 1962.

[32] J. Ba, R. Kiros, and G. Hinton, "Layer normalization," *arXiv:1607.06450*, 2016.

[33] H. Khan and B. Yener, "Learning filter widths of spectral decompositions with wavelets," in *Advances in Neural Information Processing Systems*, 2018.

[34] H. Gouk, E. Frank, B. Pfahringer, and M. Cree, "Regularisation of neural networks by enforcing Lipschitz continuity," *arXiv:1804.04368*, 2018.

[35] E. Jaynes, "Information theory and statistical mechanics," *Physical Review*, vol. 106, pp. 620–630, 1957.

[36] Y. Altun, A. Smola, and T. Hofmann, "Exponential families for conditional random fields," in *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, 2004.

[37] D. Kingma, T. Salimans, and M. Welling, "Variational dropout and the local reparameterization trick," in *Advances in Neural Information Processing Systems*, 2015.

[38] D. Molchanov, A. Ashukha, and D. Vetrov, "Variational dropout sparsifies deep neural networks," in *Proceedings of the 34th International Conference on Machine Learning*, 2017.

[39] M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. New York: Dover, 1972.

[40] J. Stoer and R. Bulirsch, *Introduction to Numerical Analysis*. Springer, 2002.

[41] S. Xiang and F. Bornemann, "On the convergence rates of Gauss and Clenshaw–Curtis quadrature for functions of limited regularity," *arXiv:1203.2445v3*, 2012.

[42] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent NNs from overfitting," *Journal of Machine Learning Research*, 2014.

[43] J. Hron, A. Matthews, and Z. Ghahramani, "Variational Bayesian dropout: pitfalls and fixes," in *Proceedings of the 35th International Conference on Machine Learning*, 2018.

[44] H. Chipman, "Bayesian variable selection with related predictors," *Canadian Journal of Statistics*, 1996.

[45] E. George and R. McCulloch, "Variable selection via Gibbs sampling," *Journal of the American Statistical Association*, vol. 88, no. 423, pp. 881–889, 1993.

[46] T. Mitchell and J. Beauchamp, "Bayesian variable selection in linear regression," *Journal of the American Statistical Association*, 1988.

[47] S. Kullback, *Information Theory and Statistics*. Wiley, 1959.

[48] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. MIT Press, 2005.

[49] M. Ager, Z. Cvetkovic, and P. Sollich, "Speech recognition front end without information loss," *arXiv:1312.6849*, 2015.

[50] Y. Hoshen, R. Weiss, and K. Wilson, "Speech acoustic modeling from raw multichannel waveforms," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015.

[51] Z. Tüske, P. Golik, R. Schlüter, and H. Ney, "Acoustic modeling with deep neural networks using raw time signal for LVCSR," in *INTERSPEECH*, 2014, pp. 890–894.

[52] S. Watanabe, Y. Minami, A. Nakamura, and N. Ueda, "Variational Bayesian estimation and clustering for speech recognition," *IEEE Transactions on Speech and Audio Processing*, 2004.

[53] S. Hu, M. Lam, X. Xie, S. Liu, J. Yu, X. Wu, X. Liu, and H. Meng, "Bayesian and Gaussian process neural networks for large vocabulary continuous speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019.

[54] X. Xie, X. Liu, T. Lee, S. Hu, and L. Wang, "BLHUC: Bayesian learning of hidden unit contributions for deep neural network speaker adaptation," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019.

[55] P. Agrawal and S. Ganapathy, "Deep variational filter learning models for speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019a.

[56] P. Agrawal and S. Ganapathy, "Modulation filter learning using deep variational networks for robust speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, 2019b.

[57] D. Palaz, R. Collobert, and M. Magimai-Doss, "Estimating phoneme class conditional probabilities from raw speech signal using convolutional neural networks," in *INTERSPEECH*, 2013.

[58] E. Loweimi, P. Bell, and S. Renals, "On learning interpretable cnns with parametric modulated kernel-based filters," in *INTERSPEECH*, 2019.

[59] N. Zeghidour, N. Usunier, I. Kokkinos, T. Schatz, G. Synnaeve, and E. Dupoux, "Learning filterbanks from raw speech for phone recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018.

[60] N. Zeghidour, N. Usunier, G. Synnaeve, R. Collobert, and E. Dupoux, "End-to-end speech recognition from the raw waveform," in *INTERSPEECH*, 2018, pp. 781–785.

[61] N. Zeghidour, Q. Xu, V. Liptchinsky, N. Usunier, G. Synnaeve, and R. Collobert, "Fully convolutional speech recognition," *arXiv:1812.06864*, 2018.

[62] W. Fisher, G. Doddington, and K. Goudie-Marshall, "The DARPA speech recognition research database: specifications and status," in *DARPA Workshop on Speech Recognition*, 1986.

[63] N. Parihar and J. Picone, "Aurora working group: DSR front end LVCSR evaluation AU/384/02," 2002.

[64] S. Renals, T. Hain, and H. Bourlard, "Recognition and interpretation of meetings: AMI and AMIDA," in *IEEE ASRU*, 2007.

[65] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi speech recognition toolkit," in *IEEE ASRU*, 2011.

[66] B. Welch, "The generalization of student's problem when several different population variances are involved," *Biometrika*, 1947.

[67] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bulletin*, 1945.

[68] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, 2006.

[69] M. Ravanelli, T. Parcollet, and Y. Bengio, "The PyTorch-Kaldi speech recognition toolkit," *arXiv:1811.07453*, 2018.

[70] M. Ravanelli, P. Brakel, M. Omologo, and Y. Bengio, "Batch-normalized joint training for dnn-based distant speech recognition," *IEEE Spoken Language Technology Workshop*, 2016.

[71] V. Peddinti, T. Sainath, S. Maymon, B. Ramabhadran, D. Nahamoo, and V. Goel, "Deep scattering spectrum with deep neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014.

[72] M. Ravanelli, P. Brakel, M. Omologo, and Y. Bengio, "Light gated recurrent units for speech recognition," *IEEE Transactions on Emerging Topics in Computing*, 2018.

[73] J. Garofolo, D. Graff, D. Paul, and D. Pallett, "CSR-I (WSJ0) Complete LDC93S6A," *Linguistic Data Consortium*, 1993.

[74] J. Rownicka, P. Bell, and S. Renals, "Multiscale octave convolutions for robust speech recognition," in *IEEE ASRU*, 2019.

[75] D. Oglic, Z. Cvetkovic, P. Bell, and S. Renals, "A deep 2D convolutional network for waveform-based speech recognition," in *INTERSPEECH*, 2020.

[76] N. Joy, D. Oglic, Z. Cvetkovic, P. Bell, and S. Renals, "Deep scattering power spectrum features for robust speech recognition," in *INTERSPEECH*, 2020.

[77] J. Fainberg, O. Klejch, E. Loweimi, P. Bell, and S. Renals, "Acoustic model adaptation from raw waveforms with SincNet," in *IEEE ASRU*, 2019.

[78] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *INTERSPEECH*, 2015.

[79] D. Povey, V. Peddinti, D. Galvez, P. Ghahremani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for ASR based on lattice-free MMI," in *INTERSPEECH*, 2016.

[80] N. Kanda, Y. Fujita, and K. Nagamatsu, "Lattice-free state-level minimum Bayes risk training of acoustic models." in *INTERSPEECH*, 2018.

[81] S. Farquhar, L. Smith, and Y. Gal, "Try depth instead of weight correlations: mean field is a less restrictive assumption for variational inference in deep networks," in *4th workshop on Bayesian Deep Learning (NeurIPS 2019)*, 2019.

[82] T. Tieleman and G. Hinton, "Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude," COURSERA: Neural Networks for Machine Learning, 2012.

[83] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proceedings of the International Conference on Learning Representations*, 2015.

[84] T. Dozat, "Incorporating Nesterov momentum into Adam," 2015.

[85] C. Sønderby, T. Raiko, L. Maaløe, S. Sønderby, and O. Winther, "Ladder variational autoencoders," in *Advances in Neural Information Processing Systems*, 2016.

[86] A. May, A. Garakani, Z. Lu, D. Guo, K. Liu, A. Bellet, L. Fan, M. Collins, D. Hsu, B. Kingsbury, M. Picheny, and F. Sha, "Kernel approximation methods for speech recognition," *Journal of Machine Learning Research*, 2019.