# LATENT REPRESENTATION LEARNING FOR ARTIFICIAL BANDWIDTH EXTENSION USING A CONDITIONAL VARIATIONAL AUTO-ENCODER

*Pramod Bachhav, Massimiliano Todisco and Nicholas Evans*

EURECOM, Sophia Antipolis, France

{bachhav,todisco,evans}@eurecom.fr

## ABSTRACT

Artificial bandwidth extension (ABE) algorithms can improve speech quality when wideband devices are used with narrowband devices or infrastructure. Most ABE solutions employ some form of memory, implying high-dimensional feature representations that increase both latency and complexity. Dimensionality reduction techniques have thus been developed to preserve efficiency. These entail the extraction of compact, low-dimensional representations that are then used with a standard regression model to estimate high-band components. Previous work shows that some form of supervision is crucial to the optimisation of dimensionality reduction techniques for ABE. This paper reports the first application of conditional variational auto-encoders (CVAEs) for supervised dimensionality reduction specifically tailored to ABE. CVAEs, form of directed, graphical models, are exploited to model higher-dimensional log-spectral data to extract the latent narrowband representations. When compared to results obtained with alternative dimensionality reduction techniques, objective and subjective assessments show that the probabilistic latent representations learned with CVAEs produce bandwidth-extended speech signals of notably better quality.

***Index Terms*—** variational auto-encoder, latent variable, artificial bandwidth extension, dimensionality reduction, speech quality

## 1. INTRODUCTION

Legacy narrowband (NB) networks and devices typically support bandwidths of 0.3-3.4kHz. In order to provide improved speech quality, today's wideband (WB) networks support bandwidths of 50Hz-7kHz. With the transition from NB to WB networks requiring significant investment [1], artificial bandwidth extension (ABE) algorithms have been developed to improve speech quality when WB devices are used with NB devices or infrastructure. ABE is used to estimate missing highband (HB) frequency components above 3.4kHz from available NB components, typically using a regression model learned from an extensive pool of WB training data.

ABE algorithms use either a classical source-filter model [2, 3] or operate directly on complex short-term spectral estimates [4–6]. In both approaches, the use of contextual information, or *memory*, leads to more reliable estimation of HB components. Some specific back-end regression models [7–9] capture memory in the form of temporal information whereas other solutions [4, 10, 11] capture memory in the front-end instead, e.g., via delta features or static features extracted from neighbouring frames. While the use of memory improves ABE performance, it implies the use of higher dimensional features and, therefore, more complex and computationally demanding ABE regression models. This is undesirable given that ABE is often required to function on battery-powered devices.

In trying to mitigate increased complexity, [12, 13] investigated the inclusion of memory through delta Mel-frequency cepstrum coefficients (MFCCs) under the constraints of fixed dimensionality. Gains in mutual information were, however, found to be offset by reconstruction artifacts involved in MFCC inversion [13]. Our own work [14] proposed an approach to include memory in the form of static features from neighbouring frames. Dimensionality reduction was used to preserve efficiency. Our subsequent work [15] showed that memory in the form of log spectral coefficients can be used to learn a compact, low dimensional feature representation for ABE using semi-supervised stacked auto-encoders (SSAE). The work presented in this paper aims to explore the use of generative modeling techniques to improve ABE performance further. The goal is to model the distribution of higher-dimensional spectral data (that includes memory) and to extract higher-level, lower-dimensional features that improve the reliability of the ABE regression model, without affecting complexity. Essentially, we seek a form of dimensionality reduction (DR) that is tailored specifically to ABE.

Probabilistic deep generative models such as variational auto-encoders (VAEs) and their conditional variant (CVAEs) are capable of modeling complex data distributions. In contrast to bottleneck features learned by stacked auto-encoders (SAEs), the latent representation is probabilistic and can be used to generate new data. Inspired by their successful use in image processing [16–18], they have become increasingly popular in numerous fields of speech processing, e.g., speech modelling and transformation [19, 20], voice conversion [21], speech synthesis [22], speech enhancement for voice activity detection [23], emotion recognition [24] and audio source separation [25].

CVAEs generate data via the combination of latent and so-called conditioning variables. The idea in the work reported in this paper is that the conditioning variable can be optimised via an auxiliary neural network in order to learn higher-level NB features, features that are tailored to the estimation of missing HB components in an ABE task. The novel contributions of this work are: (i) the first application of VAEs and CVAEs to DR for regression tasks such as ABE; (ii) the combination of CVAE with a probabilistic encoder in the form of an auxiliary neural network which derives the conditioning variable; (iii) an approach to their joint optimisation; (iv) their application to extract probabilistic NB latent representations for estimation of missing HB data in an otherwise standard ABE framework and (v) use of the proposed approach to deliver substantially improved ABE performance.

The remainder of this paper is organised as follows. Section 2 describes a baseline ABE algorithm. Section 3 explains the proposed feature extraction scheme using VAE and CVAE. Experimental work is described in Section 4 and conclusions are presented in Section 5.

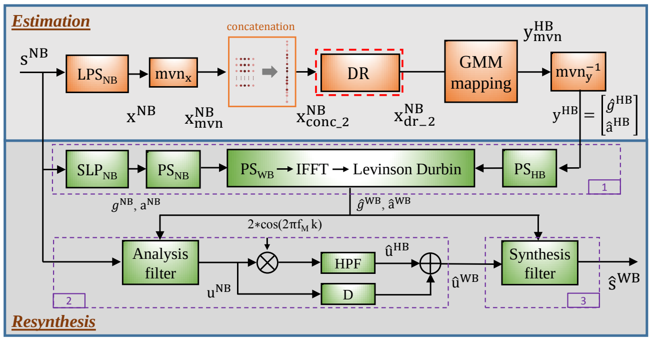

**Fig. 1**. *A block diagram of the baseline ABE system. Diagram adapted from [14].*

## 2. BASELINE ABE SYSTEM

Fig. 1 illustrates the baseline ABE system. It is identical to the source-filter model based approach presented in [14]. Accordingly, only a *brief* overview is provided here. The algorithm comprises two blocks: estimation and resynthesis.

During **estimation**, a NB speech frame $\mathbf{s}^{\text{NB}}$ of 20 ms duration with a sampling rate of 16kHz is processed using a 1024-point FFT to extract 200-dimensional NB log power spectrum (LPS$_{\text{NB}}$) coefficients $\mathbf{x}^{\text{NB}}$ which are mean and variance normalised (mvn$_{\mathbf{x}}$) to give $\mathbf{x}^{\text{NB}}_{\text{mvn}}$. After being appended with the coefficients of 2 neighbouring frames, dimensionality reduction (DR) is applied to the resulting 1000-dimensional concatenated vector $\mathbf{x}^{\text{NB}}_{\text{conc\_2}}$ to extract 10-dimensional features $\mathbf{x}^{\text{NB}}_{\text{dr\_2}}$. Normalised HB features $\mathbf{y}^{\text{HB}}_{\text{mvn}}$ consisting of 9 LP coefficients and a gain parameter are then estimated using a conventional GMM-based mapping technique [2]. Inverse mean and variance normalisation (mvn$_{\mathbf{y}}^{-1}$) is then applied, giving HB features $\mathbf{y}^{\text{HB}}$.

**Resynthesis** is performed in three steps. First (box in Fig. 1), LP parameters $\mathbf{a}^{\text{NB}}$, $g^{\text{NB}}$ are obtained from speech frame $\mathbf{s}^{\text{NB}}$ via selective linear prediction (SLP$_{\text{NB}}$) to get the NB power spectrum PS$_{\text{NB}}$. This is then concatenated with the HB power spectrum PS$_{\text{HB}}$ (obtained from estimated HB LP parameters $\hat{g}^{\text{HB}}, \hat{\mathbf{a}}^{\text{HB}}$), giving the WB power spectrum PS$_{\text{WB}}$, and hence estimated WB LP parameters $\hat{g}^{\text{WB}}, \hat{\mathbf{a}}^{\text{WB}}$. Second (box 2), the HB excitation $\hat{\mathbf{u}}^{\text{HB}}$ is estimated from the spectral translation of the NB excitation $\mathbf{u}^{\text{NB}}$ at 6.8 kHz followed by high pass filtering. NB and HB excitation components are then combined to give the extended WB excitation $\hat{\mathbf{u}}^{\text{WB}}$. Finally (box 3), $\hat{\mathbf{u}}^{\text{WB}}$ is filtered using a synthesis filter defined by $\hat{g}^{\text{WB}}$ and $\hat{\mathbf{a}}^{\text{WB}}$ in order to resynthesise speech frame $\hat{\mathbf{s}}^{\text{WB}}$. A conventional overlap and add (OLA) technique is used to produce extended WB speech.

## 3. FEATURE EXTRACTION USING CONDITIONAL VARIATIONAL AUTO-ENCODERS

We show in this section how the joint learning of VAE and CVAE architectures can be used for feature extraction in order to improve ABE performance.

### 3.1. Variational auto-encoders

A variational auto-encoder (VAE) [26] is a generative model $p_\theta(\mathbf{x}, \mathbf{z}) = p_\theta(\mathbf{z})p_\theta(\mathbf{x}|\mathbf{z})$ (with parameters $\theta$) which assumes that data $\{\mathbf{x}^{(i)}\}_{i=1}^{N}$, consisting of $N$ i.i.d. samples of a random variable $\mathbf{x}$ is generated from a continuous latent variable $\mathbf{z}$. In practice, the marginal likelihood $p_\theta(\mathbf{x})$ and true posterior density $p_\theta(\mathbf{z}|\mathbf{x})$ both are intractable. To alleviate this problem, VAEs use a recognition/inference model $q_\phi(\mathbf{z}|\mathbf{x})$ as an approximation to the posterior $p_\theta(\mathbf{z}|\mathbf{x})$. The marginal likelihood over a single datapoint is given by:

$$\log p(\mathbf{x}) = -D_{KL}[q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})] + \mathcal{L}(\theta, \phi; \mathbf{x}) \qquad (1)$$

where the first term represents the Kullback-Leibler (KL) divergence ($D_{KL}$) between the approximate and true posterior distributions. For simplicity, it is assumed that the approximate and true posteriors are diagonal multivariate Gaussian distributions whose respective parameters $\theta$ and $\phi$ are computed using two different deep neural networks.

Since the KL divergence is non-negative, $\mathcal{L}(\theta, \phi; \mathbf{x})$ represents a variational lower bound on the marginal likelihood which can be written as:

$$\mathcal{L}(\theta, \phi; \mathbf{x}) = -D_{KL}[q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z})] + \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] \quad (2)$$

where $D_{KL}[\cdot]$ acts as a regulariser which can be computed analytically. In practice, the prior $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I})$ is assumed to be a centered isotropic multivariate Gaussian with no free parameters. The second term is the expected negative reconstruction error and must be estimated by sampling. It is approximated by $\frac{1}{L}\sum_{l=1}^{L} \log p_\theta(\mathbf{x}|\mathbf{z}^{(l)})$ using $L$ samples drawn from a recognition network $q_\phi(\mathbf{z}|\mathbf{x})$. Sampling is performed using a differentiable deterministic mapping such that $\mathbf{z}^{(l)} = g_\phi(\mathbf{x}, \epsilon^{(l)}) = \mu(\mathbf{x}) + \epsilon^{(l)} \odot \sigma(\mathbf{x})$ where $\epsilon^{(l)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. $\mu_{\mathbf{z}} = \mu(\mathbf{x})$ and $\sigma_{\mathbf{z}} = \sigma(\mathbf{x})$ are outputs of the recognition network $q_\phi(\mathbf{z}|\mathbf{x})$. This is called the *reparameterization trick*. The lower bound $\mathcal{L}$ forms the objective function which can be optimized with respect to parameters $\theta$ and $\phi$ using a stochastic gradient descent algorithm.

### 3.2. Conditional variational auto-encoders

A conditional variational auto-encoder (CVAE) is a conditional, generative model, $p_\theta(\mathbf{y}, \mathbf{z}|\mathbf{x}) = p_\theta(\mathbf{z})p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{z})$. For a given input observation $\mathbf{x}$, a latent variable $\mathbf{z}$ is drawn from a prior distribution $p_\theta(\mathbf{x})$ from which the distribution $p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{z})$ generates the output $\mathbf{y}$ [17, 18]. To deal with intractability, CVAEs also use an approximate posterior $q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{y})$.

We adopt a different formulation than in [18], where we assume that the latent variable is dependent only on the output variable $\mathbf{y}$, i.e., $q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{y}) = q_\phi(\mathbf{z}|\mathbf{y})$. The variational lower bound on the conditional likelihood $p_\theta(\mathbf{y}|\mathbf{x})$ is then given by:

$$\begin{aligned}\log p_\theta(\mathbf{y}|\mathbf{x}) &\geq \mathcal{L}(\theta, \phi; \mathbf{x}, \mathbf{y}) \\ &= -D_{KL}[q_\phi(\mathbf{z}|\mathbf{y})||p_\theta(\mathbf{z})] + \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{y})}[\log p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{z})] \quad (3)\end{aligned}$$

The second term is approximated by $\frac{1}{L}\sum_{l=1}^{L} \log p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{z}^{(\mathbf{l})})$ where $\mathbf{z}^{(l)} = g_\phi(\mathbf{y}, \epsilon^{(l)}) = \mu(\mathbf{y}) + \epsilon^{(l)} \odot \sigma(\mathbf{y})$ where $\epsilon^{(l)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. $\mu_{\mathbf{z}} = \mu(\mathbf{y})$ and $\sigma_{\mathbf{z}} = \sigma(\mathbf{y})$ are outputs of the recognition network $q_\phi(\mathbf{z}|\mathbf{y})$. In practice, $L = 1$ samples are used per datapoint [26]. CVAE recognition network $q_\phi(\mathbf{z}|\mathbf{y})$ and generation network $p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{z})$ are modeled using deep neural networks.

The output distribution $p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{z})$ in Eq. 3 is chosen to be Gaussian with mean $f(\mathbf{x}, \mathbf{z}; \theta)$ and covariance matrix $\sigma^2 * I$, i.e., $p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{z}) = \mathcal{N}(f(\mathbf{x}, \mathbf{z}; \theta), \sigma^2 * I)$ where $f$ is a deterministic transformation of $\mathbf{x}$ and $\mathbf{z}$ with parameters $\theta$. Therefore,

$$\log p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{z}) = C - \|\mathbf{y} - f(\mathbf{x}, \mathbf{z}; \theta)\|^2/\alpha \qquad (4)$$

where $C$ is a constant that can be ignored during optimisation. The scalar $\alpha = 2\sigma^2$ can be seen as a weighting factor between the KL-divergence and the reconstruction term [27].

### 3.3. Extracting latent representations for ABE

This section describes the proposed scheme to jointly optimise VAE and CVAE in order to learn latent representations tailored to ABE. The scheme is illustrated in Fig. 2. Parallel training data consisting of NB and WB utterances is processed in frames of 20ms duration with 10ms overlap. Input data $\mathbf{x} = \mathbf{x}^{\text{NB}}_{\text{conc\_2}}$ consists of NB LPS coefficients with memory (as described in Section 2). The output data $\mathbf{y} = \mathbf{y}^{\text{HB}}_{\text{mvn}}$ consists of 9 LP coefficients and a gain parameter extracted from parallel HB data via selective linear prediction (SLP).

First, the VAE is trained whereby the encoder $q_{\phi_\mathbf{x}}(\mathbf{z_x}|\mathbf{x})$ (bottom of Fig. 2) is fed with input data $\mathbf{x}$ in order to predict the mean $\mu_{\mathbf{z_x}}$ and log-variance $\log(\sigma^2_{\mathbf{z_x}})$ that represent the posterior distribution $q_{\phi_\mathbf{x}}(\mathbf{z_x}|\mathbf{x})$. A corresponding decoder $p_{\theta_\mathbf{x}}(\mathbf{x}|\mathbf{z_x})$ (not shown in Fig.2) is fed with input $\mathbf{z_x} \sim q_{\phi_\mathbf{x}}(\mathbf{z_x}|\mathbf{x})$ in order to predict the mean $\mu_\mathbf{x}$ of the distribution $p_{\theta_\mathbf{x}}(\mathbf{x}|\mathbf{z_x})$. This can be considered as some form of pretraining to initialise weights of the encoder $q_{\phi_\mathbf{x}}(\mathbf{z_x}|\mathbf{x})$. Note that, at this stage, the NB representation $\mathbf{z_x}$ is learned without any supervision from HB data. The VAE decoder is then discarded. The encoder $q_{\phi_\mathbf{x}}(\mathbf{z_x}|\mathbf{x})$ is then used as the conditioning variable of the CVAE (as shown in Fig. 2).

The CVAE is then trained to model the distribution of the output $\mathbf{y}$ as follows. The HB data $\mathbf{y}$ is fed to the encoder $q_{\phi_\mathbf{y}}(\mathbf{z_y}|\mathbf{y})$ (top-left network in Fig. 2) in order to predict the mean $\mu_{\mathbf{z_y}}$ and log-variance $\log(\sigma^2_{\mathbf{z_y}})$ of the approximate posterior distribution $q_{\phi_\mathbf{y}}(\mathbf{z_y}|\mathbf{y})$ . The predicted parameters are then used to obtain the latent representation $\mathbf{z_y} \sim q_{\phi_\mathbf{y}}(\mathbf{z_y}|\mathbf{y})$ of the output variable $\mathbf{y}$ via the *reparameterization trick* (see Section 3.2). Next, the latent variable $\mathbf{z_x} \sim q_{\phi_\mathbf{x}}(\mathbf{z_x}|\mathbf{x})$, is used as the CVAE conditioning variable. After concatenation, $\mathbf{z_x}$ and $\mathbf{z_y}$ are fed to the decoder $p_{\theta_\mathbf{y}}(\mathbf{y}|\mathbf{z_x}, \mathbf{z_y})$ (top-right network) in order to predict the mean $\mu_\mathbf{y} = \mu(\mathbf{z_x}, \mathbf{z_y})$ of the ouput variable $\mathbf{y}$. Finally, the entire network is trained to learn parameters $\phi_\mathbf{x}$, $\phi_\mathbf{y}$ and $\theta_\mathbf{y}$ jointly. From Eq. 3 and 4, the equivalent variational lower bound under optimisation is given by:

$$\log p_{\theta_\mathbf{y}}(\mathbf{y}|\mathbf{z_x}) \geq \mathcal{L}(\theta_\mathbf{y}, \phi_\mathbf{y}, \phi_\mathbf{x}; \mathbf{z_x}, \mathbf{y}) = \\ -\left[D_{KL}[q_{\phi_\mathbf{y}}(\mathbf{z_y}|\mathbf{y})||p_{\theta_\mathbf{y}}(\mathbf{z_y})] + \|\mathbf{y} - f(\mathbf{z_x}, \mathbf{z_y}; \theta_\mathbf{y})\|^2/\alpha\right] \quad (5)$$

It is expected that, during optimisation of Eq. 5, parameters $\phi_\mathbf{x}$ of the encoder $q_{\phi_\mathbf{x}}(\mathbf{z_x}|\mathbf{x})$ are updated so that the framework learns the latent representation $\mathbf{z_x}$ that encodes information about the generated CVAE output $\hat{\mathbf{y}}$.

Finally, the encoder $q_{\phi_\mathbf{x}}(\mathbf{z_x}|\mathbf{x})$ (signified by the red components in Fig. 2) is used to estimate the latent representation $\mathbf{z_x}$ for every $\mathbf{x}$. The GMM regression mapping is then learned using joint vectors $\mathbf{z_x}$ and $\mathbf{y}$ [2]. During the ABE estimation phase, the DR block (red box in Fig. 1) is replaced by the encoder $q_{\phi_\mathbf{x}}(\mathbf{z_x}|\mathbf{x})$ and estimation is performed in the same manner as described in Section 2. Note that the networks $q_{\phi_\mathbf{x}}(\mathbf{z_x}|\mathbf{x})$ and $p_{\theta_\mathbf{y}}(\mathbf{y}|\mathbf{z_x}, \mathbf{z_y})$ together form a DNN, with two stochastic layers $\mathbf{z_x}$ and $\mathbf{z_y}$, that can itself be used for ABE where $\mathbf{z_y}$ is sampled from the prior distribution $p_{\theta_\mathbf{y}}(\mathbf{z_y}) = \mathcal{N}(\mathbf{0}, \mathbf{I})$ during estimation phase. However, the aim of the work reported in this paper is to use the latent representation $\mathbf{z_x}$ learned using a CVAE as a DR technique for ABE. The aim is to preserve the computational efficiency of the regression model.

## 4. EXPERIMENTAL SETUP AND RESULTS

This section describes the databases used for ABE experiments, baseline and CVAE configuration details and results. Experiments are designed to compare the performance of ABE system that use features learned from CVAE with those that use alternative DR techniques. In all cases, performance is assessed with and without mean and variance normalisation.

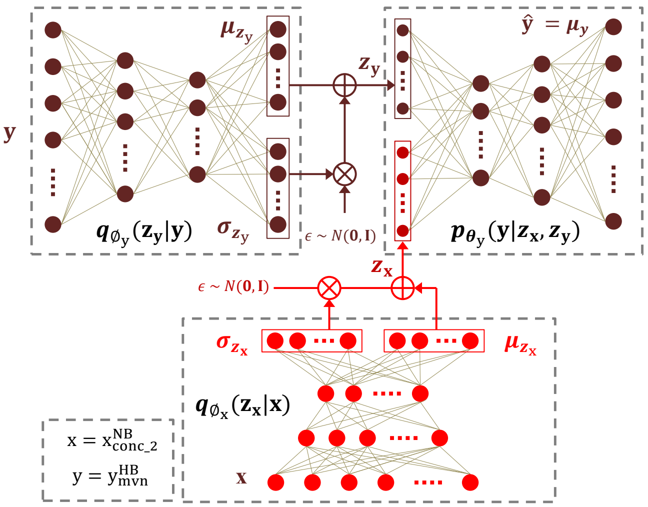

**Fig. 2**. *A feature extraction scheme using CVAE.*

### 4.1. Database

The TIMIT dataset [28] was used for training and validation. ABE solutions were trained with the 3696 utterances from the training set and 1152 utterances from the test set (excluding core test subset) using parallel WB and NB speech signals processed according to the steps described in [6]. The TIMIT core test subset (192 utterances) was used for validation and for network optimisation. The acoustically-different TSP database [29] comprising 1378 utterances was used for testing. TSP data were downsampled to 16kHz and similarly pre-processed to obtain parallel WB and NB data.

### 4.2. CVAE configuration and training

The CVAE architecture [1] is implemented using the Keras toolkit [30]. Encoders $q_{\phi_\mathbf{x}}(\mathbf{z_x}|\mathbf{x})$ and $q_{\phi_\mathbf{y}}(\mathbf{z_y}|\mathbf{y})$ consist of two hidden layers with 512 and 256 units, and 1000 and 10 units for input layers respectively. Their outputs are Gaussian-distributed latent variable layers $\mathbf{z_x}$ and $\mathbf{z_y}$ consisting of 10 units for the means $\mu_{\mathbf{z_x}}$, $\mu_{\mathbf{z_y}}$ and log-variances $\sigma_{\mathbf{z_x}}$, $\sigma_{\mathbf{z_y}}$. The decoders $p_{\theta_\mathbf{x}}(\mathbf{x}|\mathbf{z_x})$ and $p_{\theta_\mathbf{y}}(\mathbf{y}|\mathbf{z_x}, \mathbf{z_y})$ have 2 hidden layers with 256 and 512 units. Output layers have 1000 and 10 units respectively. All hidden layers have *tanh* activation units whereas Gaussian parameter layers have *linear* activation units. The modelling of log-variances avoids the estimation of negative variances.

Training is performed jointly in order to minimise the negative conditional log-likelihood in Eq. 5 using the Adam stochastic optimisation technique [31] with an initial learning rate of $10^{-3}$ and hyperparameters $\beta_1 = 0.9, \beta_2 = 0.999$ and $\epsilon = 10^{-8}$. Networks are initialised according to the approach described in [32] so as to improve the rate of convergence. To discourage over-fitting, batch-normalisation [33] is applied before every activation layer. The learning rate is reduced by half when the validation loss increases between 5 consecutive epochs. First, the VAE is trained on input data $\mathbf{x}$ for 50 epochs. The full CVAE is then trained for a further

[1] Implementation is available at `https://github.com/bachhavpramod/bandwidth_extension`

**Table 1**. *Effect of weighing factor $\alpha$ on $D_{KL}$ and RE during both training and testing phases. Results shown for the validation dataset.*

| $\alpha$ | 2 | 5 | 10 | 20 | 30 |
|---|---|---|---|---|---|
| $D_{KL}$<br>training phase | 0.96 | 0.21 | 3.3e-4 | 1.5e-4 | 9.7e-5 |
| RE<br>training phase | 4.73 | 7.40 | 8.93 | 8.97 | 8.97 |
| RE<br>testing phase | 11.40 | 9.85 | 8.93 | 8.97 | 8.97 |

**Table 2**. *Objective assessment results. RMS-LSD and $d_{COSH}$ are distance measures (lower values indicate better performance) in dB, whereas MOS-LQO$_{\mathrm{WB}}$ values reflect quality (higher values indicate better performance).*

| DR method | $d_{\text{RMS-LSD (dB)}}$ | $d_{\text{COSH (dB)}}$ | MOS-LQO$_{\mathrm{WB}}$ |
|---|---|---|---|
| PCA | 6.95 | 1.43 | 3.21 |
| PCA + MVN | 7.35 | 1.45 | 3.14 |
| SAE | 12.45 | 2.96 | 1.95 |
| SAE + MVN | 7.54 | 1.50 | 3.03 |
| VAE | 8.64 | 1.67 | 2.75 |
| VAE + MVN | 8.60 | 1.67 | 2.75 |
| SSAE | 10.50 | 2.11 | 2.26 |
| SSAE + MVN | 6.80 | 1.34 | 3.28 |
| CVAE | **6.59** | **1.31** | **3.34** |
| CVAE + MVN | **6.69** | **1.30** | **3.31** |

50 epochs using input $\mathbf{x}$ and output $\mathbf{y}$ data. The model giving the lowest validation loss is used for subsequent processing.

CVAE performance is compared to alternative SAE, SSAE and PCA DR techniques. In accordance with our previous work [15], SSAE and SAE setups have a common structure of (512, 256, 10, 256, 512) hidden units. The parameters were chosen based on our investigations in [15].

### 4.3. Analysis of weighting factor $\alpha$

Since better estimation of HB components is crucial to ABE performance, the latent representation $\mathbf{z_x}$ should contain information that is informative of $\mathbf{y}$. We therefore studied the importance of the weighing factor $\alpha$ on the reconstruction error (RE), $\|\mathbf{y} - f(\mathbf{z_y}, \mathbf{z_x}; \theta_{\mathbf{y}})\|^2$ , both during *training* and *testing* phases.

Table 1 shows the $D_{KL}$ and RE values at the end of epoch with the smallest validation loss for different values of $\alpha$. Lower values of $\alpha$ lead to higher values of $D_{KL}$, suggesting that the approximate posterior $q_{\phi_{\mathbf{y}}}(\mathbf{z_y}|\mathbf{y})$ is far from the prior distribution $p_{\theta_{\mathbf{y}}}(\mathbf{z_y}) = \mathcal{N}(\mathbf{0}, \mathbf{I})$. This hypothesis is confirmed by the observation of higher REs during *testing* than during *training*. This is because the decoder $p_{\theta_{\mathbf{y}}}(\mathbf{y}|\mathbf{z_x}, \mathbf{z_y})$ reconstructs the output $\mathbf{y}$ using latent variables $\mathbf{z_y}$ sampled from the prior during *testing*, but from the approximate posterior during *training*. Higher values of $\alpha$ give lower values of $D_{KL}$, suggesting that the posterior distribution is closer to the prior distribution. This hypothesis is confirmed by the observation of similar REs for *training* and *testing* phases. These findings corroborate those of previous work [20]. Based upon REs for the validation dataset, all experiments reported in the remainder of this paper correspond to a value of $\alpha = 10$.

**Table 3**. *Subjective assessment results for the ABE systems with CVAE, SSAE + MVN and PCA DR techniques in terms of CMOS.*

| Comparison A → B | CMOS |
|---|---|
| CVAE → NB | 0.90 |
| CVAE → PCA | 0.13 |
| CVAE → SSAE + MVN | 0.10 |
| CVAE → WB | -0.96 |

### 4.4. Objective assessment

Objective spectral distortion measures include: the root mean square log-spectral distortion (RMS-LSD); the so-called COSH measure (symmetric version of the Ikatura-Saito distortion) [34] calculated for a frequency range 3.4-8kHz, and a WB extension to the perceptual evaluation of speech quality algorithm [35]. The latter gives objective estimates of mean opinion scores (MOS-LQO$_{\mathrm{WB}}$).

Results are presented in Table 2. ABE performance with PCA dimensionality reduction outperforms that with SAE and VAE techniques, signifying the importance of supervised learning or so-called discriminative fine tuning during feature extraction. While MVN degrades performance for the PCA ABE system, it improves performance for SAE and SSAE techniques significantly. The CVAE ABE system is the best performing of all and, interestingly, performance is stable with and without MVN. This is perhaps due to the *probabilistic* learning of latent representations.

### 4.5. Subjective assessment

The results of comparative, subjective listening tests are illustrated in Table 3 in the form of the comparison mean-opinion score (CMOS). Tests were performed by 15 listeners who were asked to compare the quality of 12 pairs of speech signals $A$ and $B$ listened to using DT 770 PRO headphones. They were asked to rate the quality of signal $A$ with respect to $B$ on the scale of -3 (much worse) to 3 (much better) with steps of 1. All speech files used for subjective tests are available online[2].

Speech files whose bandwidth is extended using the proposed CVAE approach were judged to be of superior quality to original NB signals (a CMOS of 0.90) though still inferior to original WB signals (a CMOS of -0.96). However, the CVAE system produces speech of better quality than alternative systems with CMOS of 0.13 and 0.10.

## 5. CONCLUSIONS

Conditional variational auto-encoders (CVAE) are directed graphical models that are used for generative modelling. This paper presents their first application to dimensionality reduction for computationally efficient artificial bandwidth extension (ABE). When used with a standard ABE regression model, the probabilistic, latent representation produced using the proposed approach does not need any post-processing such as mean and variance normalisation. The ABE system reported in this paper produces speech of substantially better quality, a result confirmed by both objective and subjective assessment. Improvements are attributed to the better modelling of high-dimensional spectral coefficients using CVAE. Crucially, they are achieved without augmenting the complexity of the regression model. Future work should compare or combine CVAEs with other generative models such as adversarial networks.

[2] http://audio.eurecom.fr/content/media

## 6. REFERENCES

[1] S. Li, S. Villette, P. Ramadas, and D. J. Sinder, "Speech bandwidth extension using generative adversarial networks," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5029–5033.

[2] K.-Y. Park and H. Kim, "Narrowband to wideband conversion of speech using GMM based transformation," in *Proc. of IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2000, pp. 1843–1846.

[3] P. Jax and P. Vary, "On artificial bandwidth extension of telephone speech," *Signal Processing*, vol. 83, no. 8, pp. 1707–1719, 2003.

[4] K. Li and C.-H. Lee, "A deep neural network approach to speech bandwidth expansion," in *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4395–4399.

[5] R. Peharz, G. Kapeller, P. Mowlaee, and F. Pernkopf, "Modeling speech with sum-product networks: Application to bandwidth extension," in *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, 2014, pp. 3699–3703.

[6] P. Bachhav, M. Todisco, M. Mossi, C. Beaugeant, and N. Evans, "Artificial bandwidth extension using the constant Q transform," in *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5550–5554.

[7] I. Katsir, D. Malah, and I. Cohen, "Evaluation of a speech bandwidth extension algorithm based on vocal tract shape estimation," in *Proc. of Int. Workshop on Acoustic Signal Enhancement (IWAENC)*. VDE, 2012, pp. 1–4.

[8] Y. Gu, Z.-H. Ling, and L.-R. Dai, "Speech bandwidth extension using bottleneck features and deep recurrent neural networks.," in *Proc. of INTERSPEECH*, 2016, pp. 297–301.

[9] Y. Wang, S. Zhao, J. Li, J. Kuang, and Q. Zhu, "Recurrent neural network for spectral mapping in speech bandwidth extension," in *Proc. of IEEE Global Conf. on Signal and Information Processing (GlobalSIP)*, 2016, pp. 242–246.

[10] B. Liu, J. Tao, Z. Wen, Y. Li, and D. Bukhari, "A novel method of artificial bandwidth extension using deep architecture," in *Sixteenth Annual Conf. of the Int. Speech Communication Association*, 2015.

[11] J. Abel, M. Strake, and T. Fingscheidt, "Artificial bandwidth extension using deep neural networks for spectral envelope estimation," in *Proc. of Int. Workshop on Acoustic Signal Enhancement (IWAENC)*. IEEE, 2016, pp. 1–5.

[12] A. Nour-Eldin and P. Kabal, "Objective analysis of the effect of memory inclusion on bandwidth extension of narrowband speech," in *Proc. of INTERSPEECH*, 2007, pp. 2489–2492.

[13] A. Nour-Eldin, "Quantifying and exploiting speech memory for the improvement of narrowband speech bandwidth extension," Ph.D. Thesis, McGill University, Canada, 2013.

[14] P. Bachhav, M. Todisco, and N. Evans, "Exploiting explicit memory inclusion for artificial bandwidth extension," in *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5459–5463.

[15] P. Bachhav, M. Todisco, and N. Evans, "Artificial bandwidth extension with memory inclusion using semi-supervised stacked auto-encoders," in *Proc. of INTERSPEECH*, 2018, pp. 1185–1189.

[16] D. Kingma et al., "Semi-supervised learning with deep generative models," in *Advances in Neural Information Processing Systems*, 2014, pp. 3581–3589.

[17] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," in *Advances in Neural Information Processing Systems*, 2015, pp. 3483–3491.

[18] X. Yan, J. Yang, K. Sohn, and H. Lee, "Attribute2image: Conditional image generation from visual attributes," in *European Conference on Computer Vision*. Springer, 2016, pp. 776–791.

[19] W.-N. Hsu, Y. Zhang, and J. Glass, "Learning latent representations for speech generation and transformation," *INTERSPEECH*, 2017.

[20] M. Blaauw and J. Bonada, "Modeling and transforming speech using variational autoencoders.," in *INTERSPEECH*, 2016, pp. 1770–1774.

[21] C.-C. Hsu et al., "Voice conversion from non-parallel corpora using variational auto-encoder," in *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2016 Asia-Pacific*. IEEE, 2016, pp. 1–6.

[22] K. Akuzawa, Y. Iwasawa, and Y. Matsuo, "Expressive speech synthesis via modeling expressions with variational autoencoder," *INTERSPEECH*, 2018.

[23] Y. Jung, Y. Kim, Y. Choi, and H. Kim, "Joint learning using denoising variational autoencoders for voice activity detection," *Proc. Interspeech 2018*, pp. 1210–1214, 2018.

[24] S. Latif, R. Rana, J. Qadir, and J. Epps, "Variational autoencoders for learning latent representations of speech emotion," *INTERSPEECH*, 2018.

[25] L. Pandey, A. Kumar, and V. Namboodiri, "Monoaural audio source separation using variational autoencoders," *Proc. Interspeech 2018*, pp. 3489–3493, 2018.

[26] D. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[27] C. Doersch, "Tutorial on variational autoencoders," *arXiv preprint arXiv:1606.05908*, 2016.

[28] J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, and D. Pallett, "DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1," *NASA STI/Recon technical report N*, vol. 93, 1993.

[29] P. Kabal, "TSP speech database," *McGill University, Database Version : 1.0*, pp. 02–10, 2002.

[30] F. Chollet et al., "Keras," `https://github.com/keras-team/keras`, 2015.

[31] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[32] K. He et al., "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proc. of the IEEE int. conf. on computer vision*, 2015, pp. 1026–1034.

[33] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Int. conf. on machine learning*, 2015, pp. 448–456.

[34] R. Gray, A. Buzo, A. Gray, and Y. Matsuyama, "Distortion measures for speech processing," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 28, no. 4, pp. 367–376, 1980.

[35] "ITU-T Recommendation P.862.2 : Wideband extension to Recommendation P.862 for the assessment of wideband telephone networks and speech codecs," *ITU*, 2005.